AFTAC Project No. VT/9707

Alexandria, Va. 22314

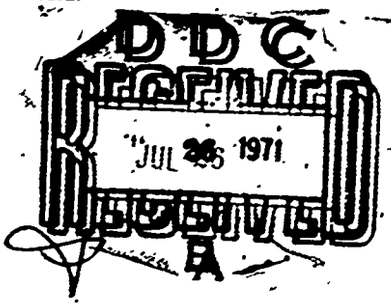
RAYLEIGH-WAVE MULTIPATH ANALYSIS USING A COMPLEX CEPSTRUM TECHNIQUE

SPECIAL REPORT NO. 2
LONG-PERIOD ARRAY PROCESSING DEVELOPMENT

Prepared by A. Frank Linville

Aaron H. Booker, Project Scientist
Stanley J. Laster, Project Scientist
T.W. Harley, Program Manager
Area Code 703, 836-3882, Ext. 60

TEXAS INSTRUMENTS INCORPORATED
Services Group
P.O. Box 5621
Dallas, Texas 75222



Contract No. F33657-69-C-1063
Amount of Contract: $779,850
Beginning 21 April 1969
Ending 31 March 1971

Prepared for
AIR FORCE TECHNICAL APPLICATIONS CENTER
Washington, D.C. 20333

Sponsored by
ADVANCED RESEARCH PROJECTS AGENCY
Nuclear Monitoring Research Office
ARPA Order No. 624
ARPA Program Code No. 9F10

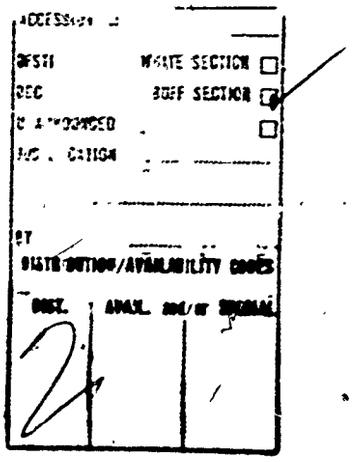
15 April 1971

Acknowledgement: This research was supported by the Advanced Research Projects Agency, Nuclear Monitoring Research Office, under Project VELA-UNIFORM, and accomplished under the technical direction of the Air Force Technical Applications Center under Contract No. F33657-69-C-1063.

This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of Chief, AFTAC.

Qualified users may request copies of this document from:

Defense Documentation Center
Cameron Station
Alexandria, Virginia 22314



Unclassified

Security Classification

DOCUMENT CONTROL DATA – R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Texas Instruments Incorporated Services Group P. O. Box 5621, Dallas, Texas 75222	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP None

3. REPORT TITLE

Rayleigh-Wave Multipath Analysis Using a Complex Cepstrum Technique

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Special Report No. 2

5. AUTHOR(S) (First name, middle initial, last name)

A. Frank Linville

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
15 April 1971	70	7

8a. CONTRACT OR GRANT NO. Contract No. F33657-69-C-1063 b. PROJECT NO. AFTAC Project No. VT/9707 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) N/A 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) N/A

10. DISTRIBUTION STATEMENT

This document is subject to special export controls and each transmittal to foreign nations may be made only with prior approval of Chief, AFTAC.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
ARPA Order No. 624	Air Force Technical Applications Center VELA Seismological Center Headquarters USAF, Alexandria, Va.

13. ABSTRACT

Homomorphic deconvolution is applied to separating the components of a convolution in which multipath Rayleigh-wave propagation is modeled as the convolution of a signal S with a multipath operator m. The technique is applied to a number of synthetic waveforms with fairly simple multisource and/or multipath Rayleigh-wave characteristics to demonstrate the separation obtainable between S and m. The technique is also applied to several events recorded at LASA and ALPA, and results of the method as applied to actual Rayleigh-wave recordings are presented.

DD FORM 1 NOV 65 1473

Unclassified

Security Classification

Unclassified

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Rayleigh Waves						
Multipath Propagation						
Complex Cepstrum Technique						
Linear Filtering						
Signal Estimate						
Multipath Operator						
LASA						
ALPA						

Security Classification

AFTAC Project No. VT/9707

RAYLEIGH-WAVE MULTIPATH ANALYSIS USING A COMPLEX CEPSTRUM TECHNIQUE

SPECIAL REPORT NO. 2
LONG-PERIOD ARRAY PROCESSING DEVELOPMENT

Prepared by A. Frank Linville

Aaron H. Booker, Project Scientist
Stanley J. Laster, Project Scientist
T. W. Harley, Program Manager
Area Code 703, 836-3882, Ext. 60

TEXAS INSTRUMENTS INCORPORATED
Services Group
P. O. Box 5621
Dallas, Texas 75222

Contract No. F33657-69-C-1063
Amount of Contract: $779,850
Beginning 21 April 1969
Ending 31 March 1971

Prepared for
AIR FORCE TECHNICAL APPLICATIONS CENTER
Washington, D. C. 20333

Sponsored by
ADVANCED RESEARCH PROJECTS AGENCY
Nuclear Monitoring Research Office
ARPA Order No. 624
ARPA Program Code No. 9F10

15 April 1971

Acknowledgement: This research was supported by the Advanced Research Projects Agency, Nuclear Monitoring Research Office, under Project VELA-UNIFORM, and accomplished under the technical direction of the Air Force Technical Applications Center under Contract No. F33657-69-C-1063.

This page is intentionally blank

SUMMARY

Homomorphic deconvolution is applied to separating the components of a convolution in which multipath Rayleigh-wave propagation is modeled as the convolution of a signal S with a multipath operator m. The technique is applied to a number of synthetic waveforms with fairly simple multisource and/or multipath Rayleigh-wave characteristics to demonstrate the separation obtainable between S and m. The technique is also applied to several events recorded at LASA and ALPA and results of the method as applied to actual Rayleigh-wave recordings are presented.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (CONT.)

LIST OF TABLES

RAYLEIGH-WAVE MULTIPATH ANALYSIS USING A COMPLEX CEPSTRUM TECHNIQUE

SECTION I
INTRODUCTION

The excitation and propagation of Rayleigh waves through plane multilayered media is well understood. Early studies of Rayleigh-wave recordings yielded information relating to the average crustal structure between the epicenter and the recording site by group velocity measurement. That analysis allowed definition of gross structural differences between oceanic and continental crusts. More recently, phase-velocity measurements have been used to determine crustal properties beneath a given recording site. If the dispersed Rayleigh wave propagates along the great circle path from epicenter to recording site, then only two recording stations at that site are needed for phase velocity determination. The fact that Rayleigh waves do not necessarily propagate along the great circle path but may be laterally refracted through higher velocity structure and refracted at continental margins which mark the boundary between oceanic and continental crusts was well established by Evernden (1953, 1954). A tripartite array is then needed to accurately determine the direction of approach and the true phase velocity across the array. Studies by Knopoff, et al. (1966) and Texas Instruments Incorporated (1967) using Fourier transform techniques for calculating phase velocity and direction of approach substantiates the deviation of direction of approach of Rayleigh waves from the great circle path for many recorded events and in addition shows the direction of approach to be frequency dependent.

Lateral refraction, as well as reflection and refraction at crustal transition zones along different propagation paths, yields complex interference patterns of the recorded Rayleigh waves. Recorded waveforms often appear to be amplitude modulated and studies by Pilant and Knopoff (1964) show this may occur because of a temporal and/or spatial distribution at the source or

because of multipath propagation. At periods which correspond to beats, where a minimum of energy is observed in the recording, the amplitude spectra show large minima and irregularities in the smoothness of the phase spectra occur. These phase irregularities produce large scatter in computed phase velocities and special techniques must be used to reduce the influence of beats.

Frequency-wavenumber analysis by Capon (1970) yields additional evidence of multipath propagation which is frequency dependent and conjectures were made concerning the actual paths taken by the various frequency groups arriving simultaneously at LASA during successive 200-sec time intervals over the Rayleigh wave duration.

Measurements made from the recorded Rayleigh waveform to determine group velocity, phase velocity and source depth are strongly influenced by multipath propagation yielding results which are difficult to interpret. In addition, various discriminants calculated from the contaminated Rayleigh wave (spectral ratio, MS, AR, RMS, Mchirp, etc.) are not representative of the Rayleigh wave excitation at the source.

The analysis presented here then is concerned with recovering the signal and the multipath operator from the recorded Rayleigh waveform. Homomorphic deconvolution using a generalized concept of linear filtering is applied to separating the components of a convolution in which multipath propagation is modeled as the convolution of a signal with a multipath operator. The signal would be more representative of the "pure" Rayleigh wave and the multipath operator could be interpreted in terms of differential dispersion and/or multiple sources.

SECTION II
TECHNIQUE

Schafer (1969) has presented a new approach to separating convolved signals. The approach attempts separation using a "complex cepstrum" technique. He gives a detailed analysis of homomorphic systems for deconvolution and applies the technique to a class of signals in which one of the members of the convolution is an impulse train. The technique is conceptually simple and is summarized as follows:

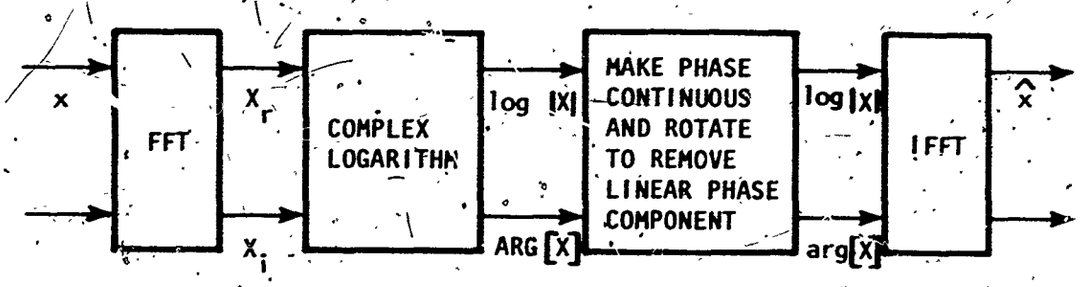
- The problem here is to estimate a signal S(t) which has been complicated by a few multipaths m(t) (this is the multipath operator). Mathematically the problem is $X(t) = S(t) * m(t)$, where * is convolution and X(t) is the observed waveform.
- The z or Fourier transform is taken and thus, $X(z) = S(z)\, m(z)$.
- The log of X(z) is taken and the imaginary part is made continuous and thus, $\log X(z) = \log S(z) + \log m(z)$.
- Log X(z) is treated as a complex time series and a linear filter is used to separate log S(z) and log m(z). Basically, this requires that the spectrum of log S(z) and log m(z) not overlap greatly.
- The exponential is taken to get back to an estimate of S(z) or m(z). There is no ambiguity here as exponentiation will map any one of the multivalued log function values back into the same complex number.
- Inverse transform to get an estimate of S(t) or m(t).

This technique has been used successfully by Schafer in separating speech signals into their more basic components and in removing echoes from speech data. The crucial link in this technique is whether or not log S(z) and log m(z) are separable for multipath Rayleigh-wave data. The numerical procedures given by Schafer were programmed for the S/360 computer and synthetic waveforms representing fairly simple multipath Rayleigh-wave recordings were used to determine the separation obtainable. The sequence of numerical calculations involved in implementing the technique is described in the following section.

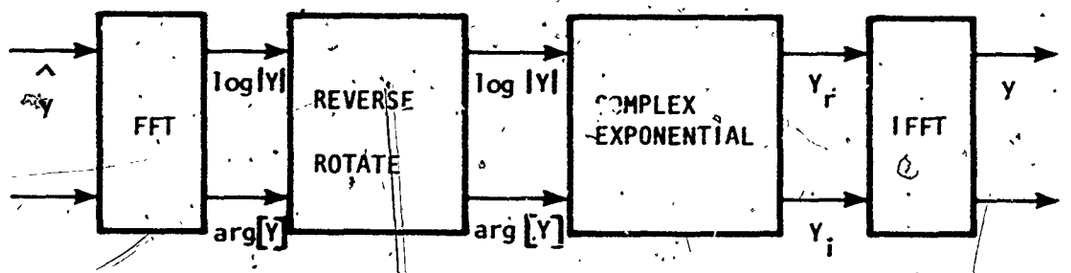
SECTION III

THEORY AND COMPUTER APPLICATION PROGRAM

Homomorphic deconvolution is based on transforming a convolution of waveforms into a sum, using a linear filter system to separate the additive components and transforming the result back to the original input space. The numerical operations performed are summarized as follows:



a linear filter is applied to the complex cepstrum $\hat{x}$ to obtain y, then,



the output y is the estimate of S(t) or m(t).

First, the fast Fourier transform (FFT) is taken

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-i \frac{2\pi}{N} kn} \qquad k = 0, 1, \ldots, N-1$$

The sign of X(o) is checked and if negative, the sign of X(k) is changed to remove the constant phase component. This is remembered and the sign of Y(k) is subsequently changed in the inverse system. The complex logarithm is taken

$$\hat{X}(k) = \log [X(k)] = \log |X(k)| + i \text{ ARG } [X(k)] \pm i2\pi q \quad k = 0, 1, \ldots, N-1$$

and the principal value ARG [X(k)] when q = 0 is obtained where

$$-\pi < \text{ARG } [X(k)] \leq \pi$$

The complex logarithm is multivalued and must be defined such that there is no ambiguity with respect to its imaginary part by choosing the integer q which makes the discrete phase values approximate a continuous phase function. This may be accomplished even for rapidly varying phase angles provided the frequency sampling is fine enough by either one of the two subroutines SCHAFR or AARON. SCHAFR uses a first difference and AARON uses a one-point prediction error to detect jumps in phase of 2π. The input signal is augmented with zeros to give an input trace N sample points in length in order to obtain the fine frequency sampling needed. The value N is a variable input parameter to the program. The estimated phase curve is then rotated by removing the linear phase component and arg [Y] is subsequently reverse rotated in the inverse system. Next the inverse fast Fourier transform (IFFT) is taken.

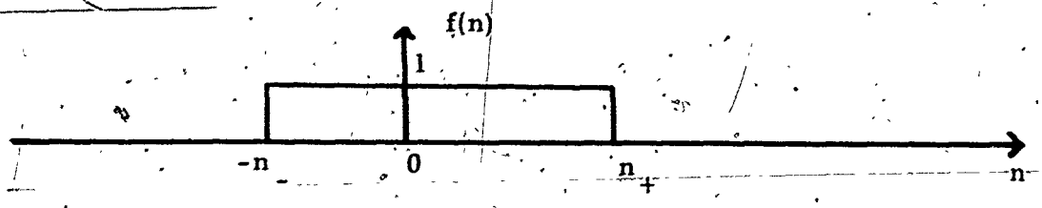
$$\hat{x}(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)\, e^{i \frac{2\pi}{N} kn} \qquad n = 0, 1, \ldots, N-1$$

yielding the complex cepstrum $\hat{x}$.

The part due to S(t) in the complex cepstrum is primarily concentrated in the "short time" region while the impulses attributed to m(t) occupy the "long time" region. If these signals are separated in "time", then a linear filter system can be used to separate the signals from one another. A short-pass or comb filter system can be used to remove the impulses due to m(t) from the complex cepstrum while retaining most of the part attributed to S(t). Similarly, a long-pass filter system can be used to remove most S(t) while

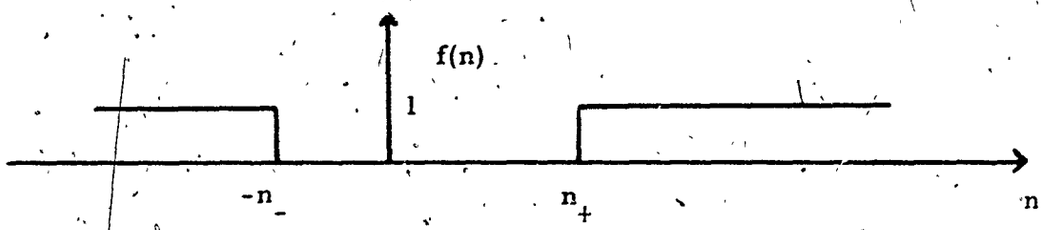
retaining the part due to m(t). The complex cepstrum $\hat{x}$, once calculated, is saved and the program recycles to read input filter parameters, apply the filter to the complex cepstrum and calculate the output of the inverse system. This allows the analyst to obtain outputs for a suite of filters for interpretive purposes. The short-pass linear filter system is defined as

$$f(n) = 1 \qquad -n_- < n < n_+$$
$$= 0 \qquad \text{elsewhere}$$



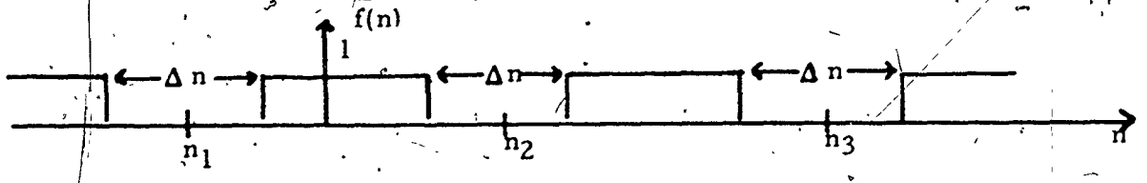
where n_- and n_+ are integer input parameters. The long-pass system is defined as

$$f(n) = 0 \qquad -n_- < n < n_+$$
$$= 1 \qquad \text{elsewhere}$$



where n_- and n_+ are integer input parameters. The comb filter system is defined as

$$f(n) = 0 \qquad n_k - \frac{\Delta n_k}{2} < n < n_k + \frac{\Delta n_k}{2} \qquad k = 1, 2, \ldots$$
$$= 1 \qquad \text{elsewhere}$$



where n_k and Δn are input parameters.

The filter systems are applied in the program by zeroing the appropriate portions of the complex cepstrum. The sharp cutoff of these filter systems will produce ripples in the transform of $\hat{y}$; therefore an option is provided to modify the filter amplitude function with a cosine taper giving smoother transitions between one and zero.

The filtered complex cepstrum $\hat{y}$ then is input to the inverse system where the FFT is taken, the imaginary part reverse rotated, the complex exponential taken, and the IFFT taken to yield the filtered output y.

Preprocessing of the input signal can also be performed by the program when necessary. First, the mean is removed and the signal is truncated at the nearest zero crossing at each end. A zero-phase bandpass filter is designed and convolved with the signal to allow decimation, the cutoff frequencies and decimation interval being variable input parameters. If the input sequence is nonminimum-phase, then exponential weighting may be applied in order to make the sequence more like a minimum phase sequence. It has been empirically observed that this will insure that the part of the complex cepstrum attributed to m will occur in the region $t \geq t_1$ where t_1 is the spacing between the first and second impulses. The program applies a truncated exponential window to the input sequence.

$$w(t) = \alpha^{t} x(t) \qquad 0 \leq t < L$$
$$= 0 \qquad \text{elsewhere}$$

where $|\alpha| < 1$, L is the length of the input window and α is a variable input parameter. Subsequent calculations are performed on the exponentially weighted input sequence, and the output of the inverse system is then unweighted using the same value of α.

In addition, the program generates CalComp plots of various functions obtained during the sequence of calculations to allow visual analysis of the results.

SECTION IV

PROGRAM TEST CASE

In order to check the programmed procedures, a simple example given by Schafer was processed through the program.

Input to the program is the sequence X whose values are:

$$X(t) = S_1(t) + S_2(t) = S_1(t) * m(t)$$

where the sequence S_1 has values

$$S_1(t) = ta^t \qquad 0 \leq t \leq 800$$

$$= 0 \qquad \text{elsewhere}$$

and the sequence S_2 has values

$$S_2(t) = \alpha S_1(t-t_1) \qquad 80 \leq t \leq = 880$$

$$= 0 \qquad \text{elsewhere}$$

and

$$m(t) = \delta(t) + \alpha \delta(t-t_1)$$

For this example, a = 0.96, α = 0.5, t_1 = 80, and the value N for the fast Fourier transform was 2048.

CalComp-plotted results of the sequence of calculations are given in Figure IV-1. Shown in the figure, proceeding from top to bottom, are $S_1(t)$, $S_2(t)$ and their sum X(t). Next shown are the real and imaginary parts of the complex logarithm of the FFT of the input X(t). The imaginary part

has been made continuous and then rotated to remove the linear component. The IFFT of the real and imaginary parts gives the complex cepstrum. For this example the input X(t) is minimum phase, after rotation, yielding a complex cepstrum that is zero for $t < 0$. The part due to $S_1(t)$ is primarily in the "short time" region from t = 0 to t = 80. The impulses due to the echo are in the "long time" region and occur at t = 80, t = 160, etc. The complex cepstrum is then short-pass filtered at sample point 74 and the FFT taken. The real part shows the effect of removing the rapidly varying component due to m(t). The imaginary part is reverse rotated, the complex exponential taken and the IFFT taken to yield the short-pass output which is the estimate of $S_1(t)$. An error trace is calculated [$S_1(t)$ minus the short-pass output] which shows that the output of the short-pass system estimates $S_1(t)$ with negligible error. Next, a long-pass filter at sample point 74 is applied to the complex cepstrum and processed as before. The real part shows the effect of removing the slowly varying component due to $S_1(t)$. The output of the long-pass system (bottom trace, Figure IV-1) yields m(t) with negligible error.

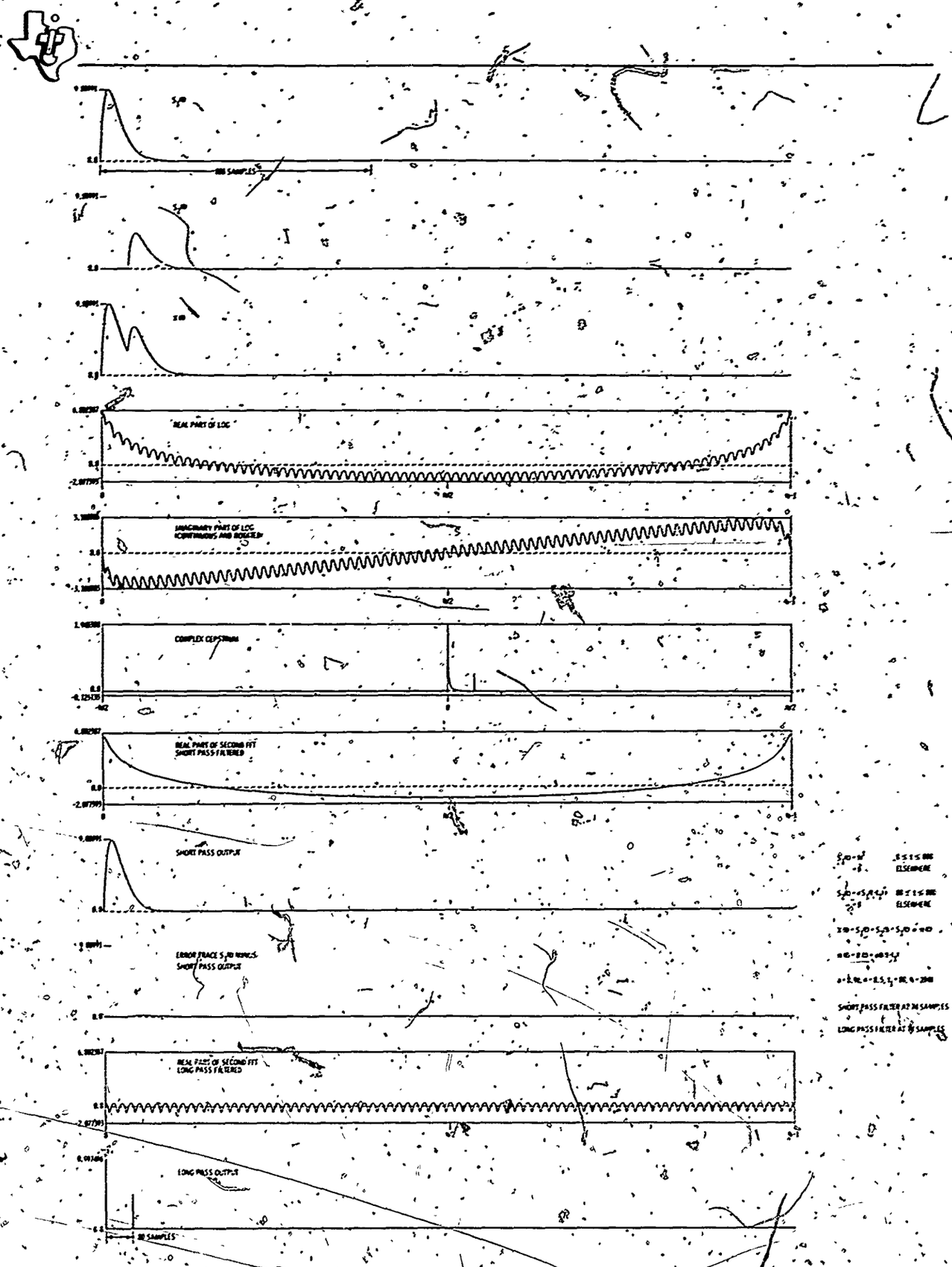


Figure IV-1. Cepstrum Results for Program Test Case

SECTION V

CEPSTRUM ANALYSIS OF SYNTHETIC MULTIPATH RAYLEIGH WAVES

Seven synthetic waveforms with simple multisource and/or multipath Rayleigh wave characteristics were processed using the complex-cepstrum technique in order to determine the separation obtainable between S(t) and m(t) and to serve as an aid in interpreting subsequent outputs obtained from actual Rayleigh wave recordings.

The waveforms used were produced by adding together two or more cosine-tapered chirp waveforms. First, a chirp waveform is generated by specifying the initial frequency f_0, final frequency f_1, and time duration L, where the impulse response is

$$R(t) = \sin\left[2\pi\left(f_0 + \frac{f_1 - f_0}{2L}\,t\right)t\right] \qquad 0 \le t \le L$$

$$= 0 \qquad \text{otherwise}$$

where

$f_0 = 0.025$ Hz and $f_1 = 0.$ [illegible] Hz

Shown in Figure V-1 (top) is a chirp waveform (R(t) with a time duration of 500 sec.

In order to make the waveform more nearly resemble an actual seismogram recording of a Rayleigh wave, a cosine taper is applied to R(t) where the output is

$$S(t) = R(t)\left[\frac{1}{2}\left(1 - \cos\frac{2\pi t}{L}\right)\right] \qquad 0 \le t \le L$$

This is analogous to shaping the spectrum with a recording system's response (long-period seismometer). The cosine-tapered chirp waveform (Figure V-1, middle) resembles an actual teleseismic seismogram recording of a Rayleigh wave in that it exhibits normal dispersion and its amplitude spectrum is highly

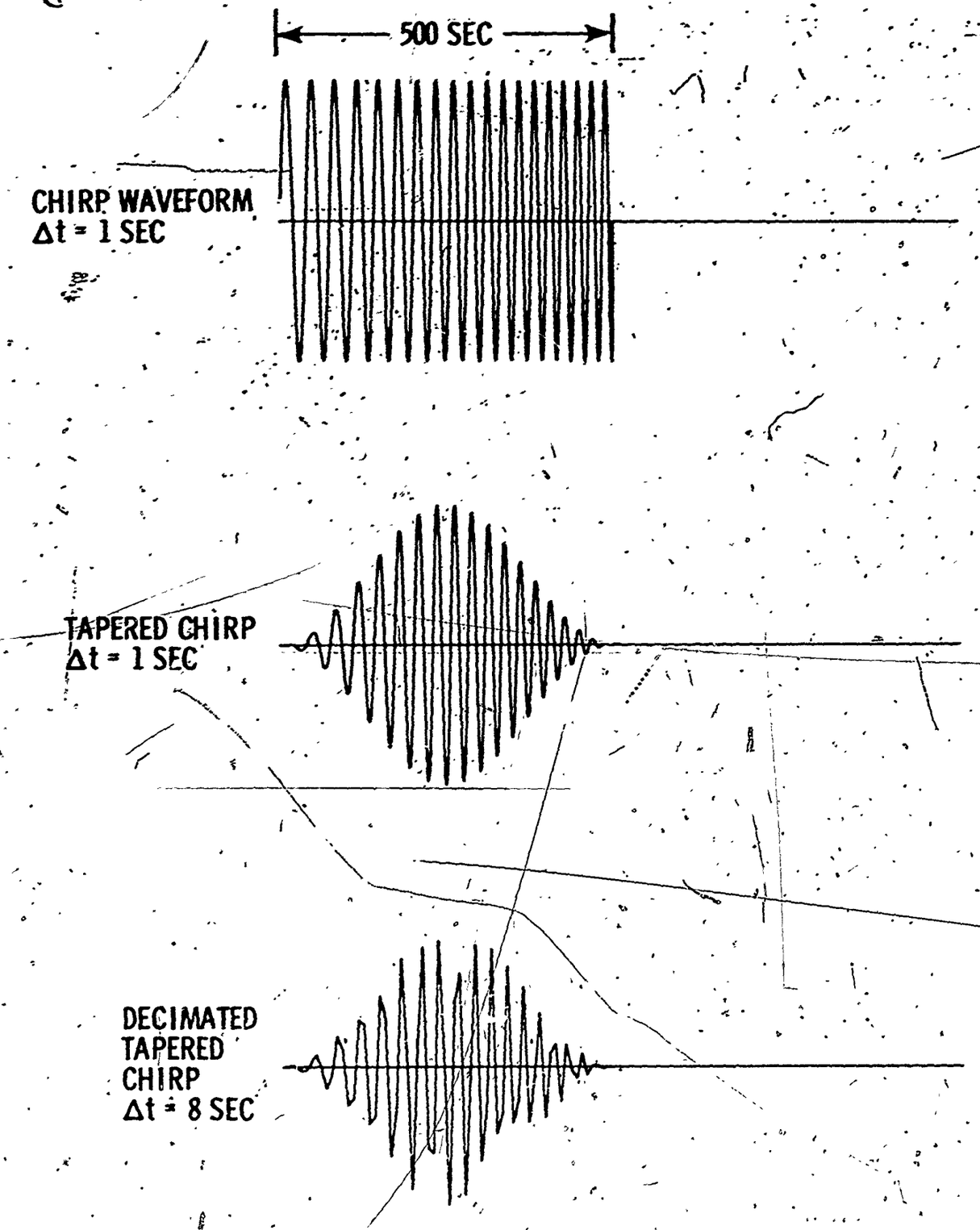


Figure V-1. Chirp Waveform Synthesis

peaked.

Application of the cepstrum technique was found empirically to work better by resampling the data at 8-sec intervals (Figure V-1, bottom). This sample period gives a Nyquist frequency of 0.0625 Hz which is well outside the frequency band specified for the chirp waveforms. As noted in Figure V-1, resampling of the waveform yields an irregular appearing time function, however, there is no loss of spectral information in the band of interest. Parameters calculated directly from the decimated time function such as group velocity (wave frequency and arrival times of peaks and troughs) and maximum amplitudes of given peaks and troughs are determined using a three-point quadratic interpolation yielding results essentially identical with those calculated from the waveform sampled at 1 sec.

The first multichirp waveform analyzed consists of a 500 sec chirp with onset at 1-sec plus a 500-sec chirp with onset at 250 sec. Shown in Figure V- are the two individual chirps, their sum, and the decimated sum which is the input signal to the complex-cepstrum analysis program. Group velocities calculated from the waveforms are presented in the right side of the figure. For reference, the group velocity curve (solid line) for the LASA perturbed model (Texas Instruments Incorporated, 1967) is included in the plots. A distance and origin time is chosen such that the resultant group velocity points calculated from $S_1(t)$ closely fit the LASA perturbed curve. For this example, the distance was 9053 km and the origin time was 2543 sec before trace onset. The multichirp waveform yields a complicated group velocity curve. The smooth points at at low frequencies and the smooth points at high frequencies derive from the non-over-lapping portions of $S_1(t)$ and $S_2(t)$ respectively. A complex interference pattern results, where $S_1(t)$ and $S_2(t)$ overlap in time yielding the scattering of points observed at intermediate frequencies. For comparison, group velocity calculations for several earthquakes (catalogued in Table VI-1) recorded at LASA are presented in Figure V-3. LASA recordings of Rayleigh waves from events occurring in many widely separated regions yield complicated group velocity curves with characteristics very similar to the multichirp examples presented in this section.

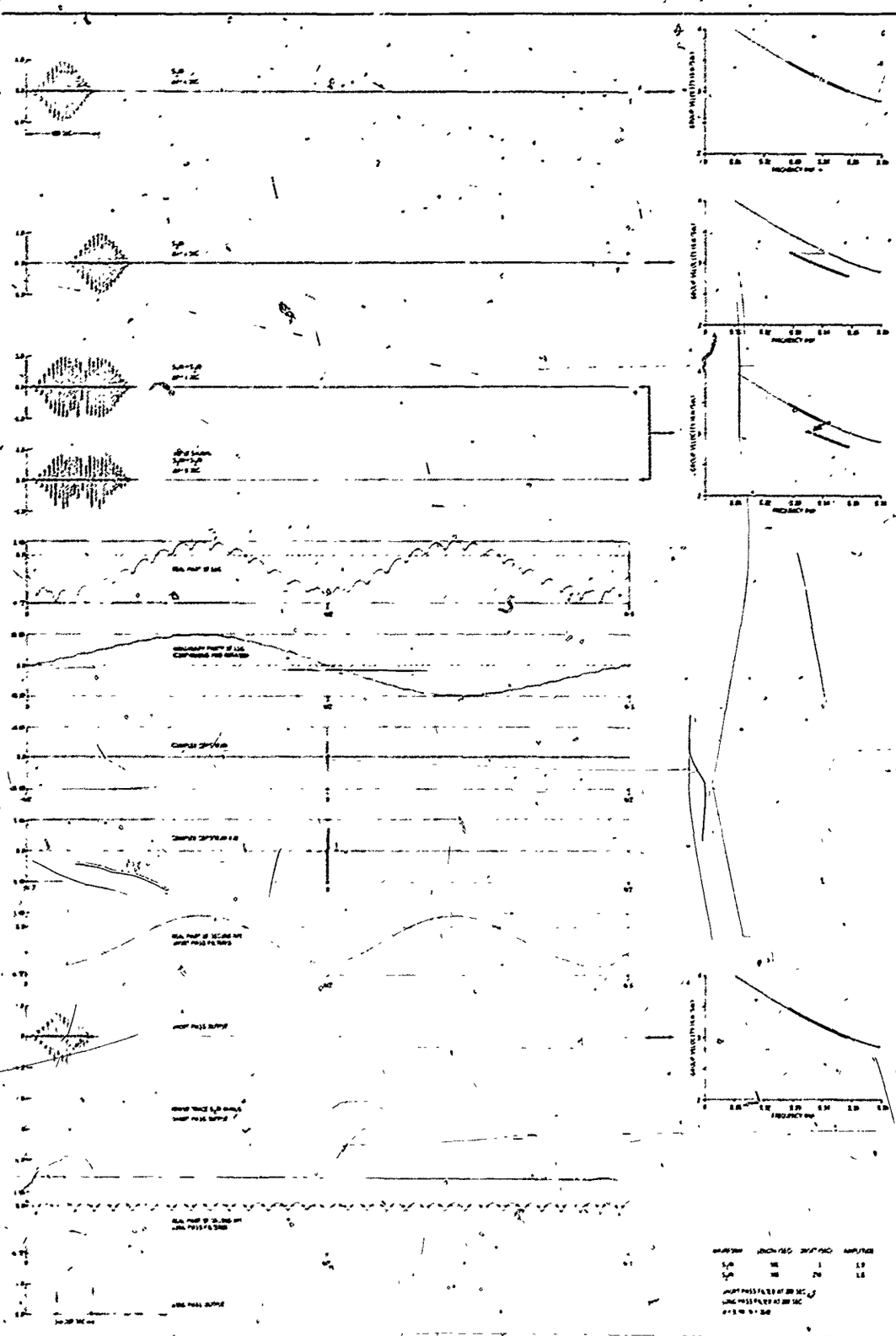


Figure V-2. Cepstrum Results for Synthetic Multipath Rayleigh Waves. Input is the Sum of Two Chirp Waveforms; 500 Sec and 500 Sec in Length with Onsets at 1 Sec and 250 Sec and Amplitudes of 1 Unit and 1 Unit

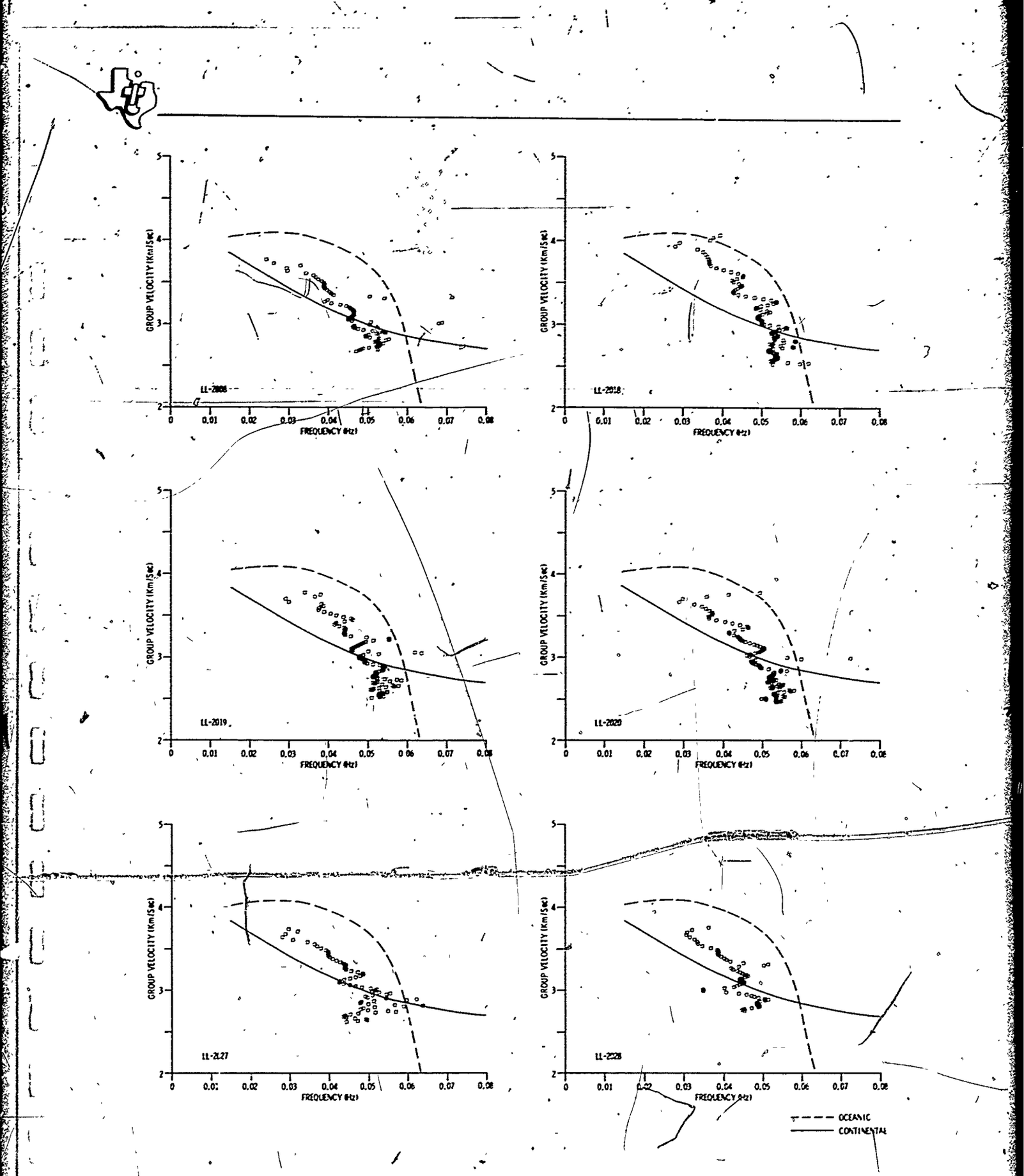


Figure V-3. Group Velocity vs Frequency Plots

The multichirp waveforms are not minimum phase, therefore exponential weighting is applied to the input signal and subsequent calculations are performed on the exponentially weighted sequence. The value for α used in the weighting was 0.98 for this example, and the output of the system is then unweighted using the same value for α.

Results of the sequence of calculations using the complex-cepstrum technique are presented in the remaining portion of Figure V-2. Shown are the real and imaginary parts of the complex logarithm of the FFT of the exponentially weighted input signal. The slowly varying component due to S and the rapidly varying component due to m are clearly seen in the plot of the real part of the log. The value N given in the figures is the number of samples for the FFT and is a power of two. The input signal is augmented with zeros before transformation to give a trace N-samples long. For the frequency-domain plots, N is the number of frequency samples and the folding frequnecy w is a N/2. For a sample period $\Delta t = 8$ sec, the folding frequency is

$$w = \frac{1}{2\Delta t} = 0.0625 \text{ Hz}$$

and for the time domain plots (complex cepstrum)

$$N/2 = N\Delta t/2 = 8192 \text{ sec}$$

The complex cepstrum is shown next. The impulses attributed to m are small in amplitude relative to the pulse attributed to S which occurs primarily in the "short time" region; therefore an enlarged plot of the complex cepstrum is also shown. The large pulse about zero time is clipped in the enlarged plot for display purposes. The impulses attributed to m occur at $t = 249$, $t = 498$, etc. and are easily discerned in the enlarged plot.

The complex cepstrum was short-pass filtered at 208 sec, the second FFT taken, the imaginary part reverse rotated, the complex exponential taken, the second IFFT taken, and the sequence unweighted to yield the short-pass

output which is the estimate of $S_1(t)$. The error trace shows that the output of the short-pass system estimates $S_1(t)$ with little error. Group velocity calculated from the short-pass output is shown to the right of the time trace.

Next the complex cepstrum was long-pass filtered at 208 sec and processed as before. The real part of the second FFT shows the effect of removing the slowly varying component due to S. The output of the long-pass system yields m(t) which is essentially $\delta(t) + \delta(t-249)$ for this example.

The second multichirp waveform analyzed consists of a 600-sec chirp with onset at 1 sec plus a 600-sec chirp with onset at 250 sec. Results of the sequence of calculations are shown in Figure V-4. The format of the plots in this figure and succeeding figures in this section follows that of the first example given. $S_1(t)$ and $S_2(t)$ are each 100 sec longer and overlap 100 sec more in time than for the first example. The short-pass output yields $S_1(t)$ with practically no error and, as seen in the figure, the long-pass output m(t) is essentially identical with the first example.

The third multichirp waveform analyzed consists of a 500-sec chirp with onset at 1 sec plus a 600-sec chirp with onset at 250 sec. The analysis results are presented in Figure V-5. The short-pass output again yields $S_1(t)$ with negligible error. The long-pass output m(t) is more complex than the first two examples where a constant time delay existed between $S_1(t)$ and $S_2(t)$ for all frequencies. For this example, m(t) can be interpreted in terms of differential dispersion wherein the time delay between $S_1(t)$ and $S_2(t)$ is 249 sec at 0.025 Hz and increases with increasing frequency to 349 sec at 0.050 Hz.

The fourth multichirp waveform analyzed is more complex in that it consists of three chirp waveforms 500 sec, 550 sec and 600 sec in length with onsets at 1 sec, 201 sec and 401 sec. The analysis results are presented in Figure V-6. The complex cepstrum was short-pass filtered at 160 sec and the output yields $S_1(t)$ with little error. The long-pass output m(t) is more complex than the preceding example in that it shows the differential dispersion between $S_1(t)$ and $S_2(t)$ as well as between $S_1(t)$ and $S_3(t)$. As noted in the plot

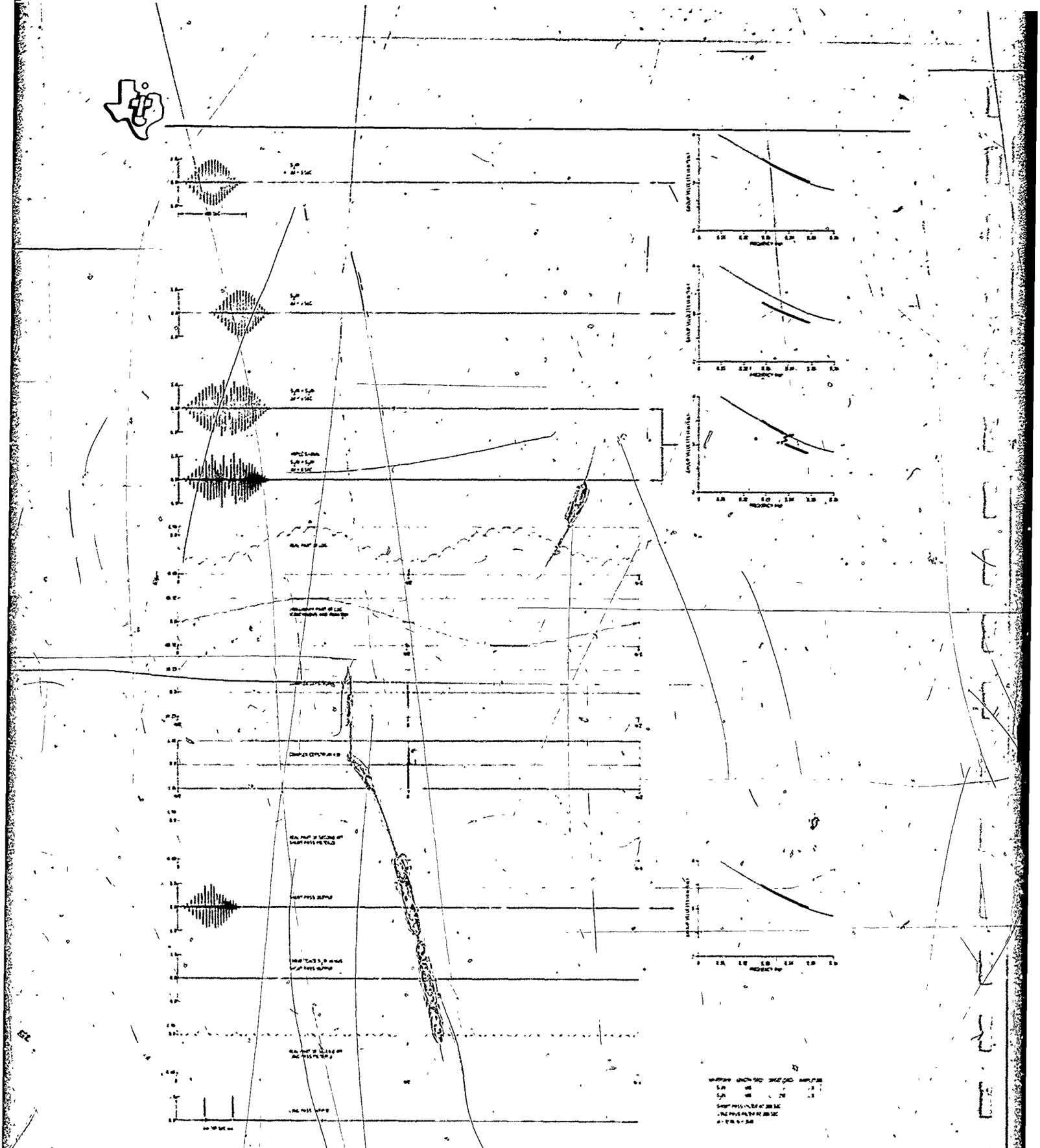


Figure V-4. Cepstrum Results for Synthetic Multipath Rayleigh Waves. Input Is the Sum of Two Chirp Waveforms; 600 Sec and 600 Sec in Length with Onsets at 1 Sec and 250 Sec and Amplitudes of 1 Unit and 1 Unit

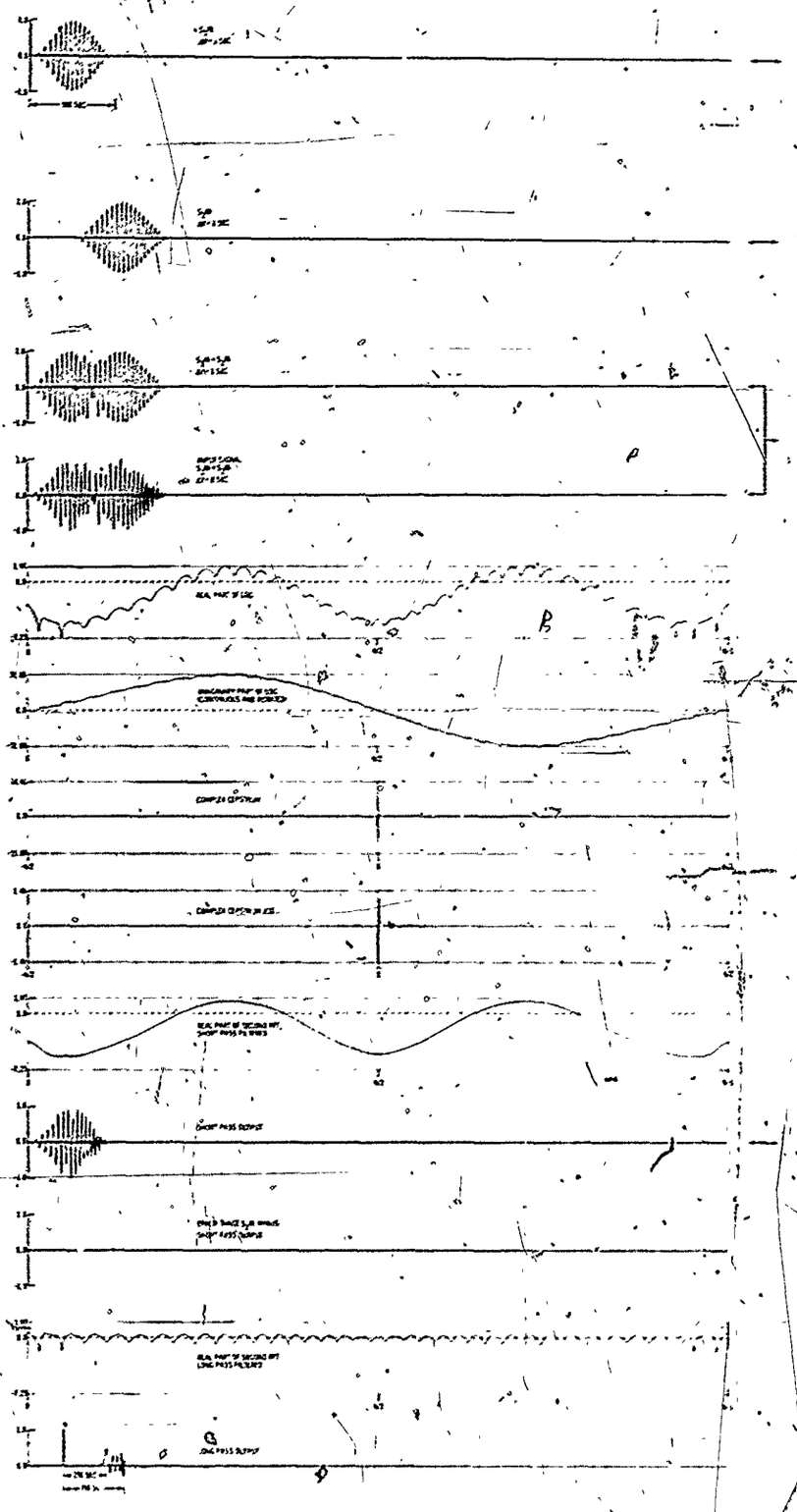


Figure V-5. Cepstrum Results for Synthetic Multipath Rayleigh Waves. Input Is the Sum of Two Chirp Waveforms; 500 Sec and 600 Sec in Length with Onsets at 1 Sec and 250 Sec and Amplitudes of 1 Unit and 1 Unit.

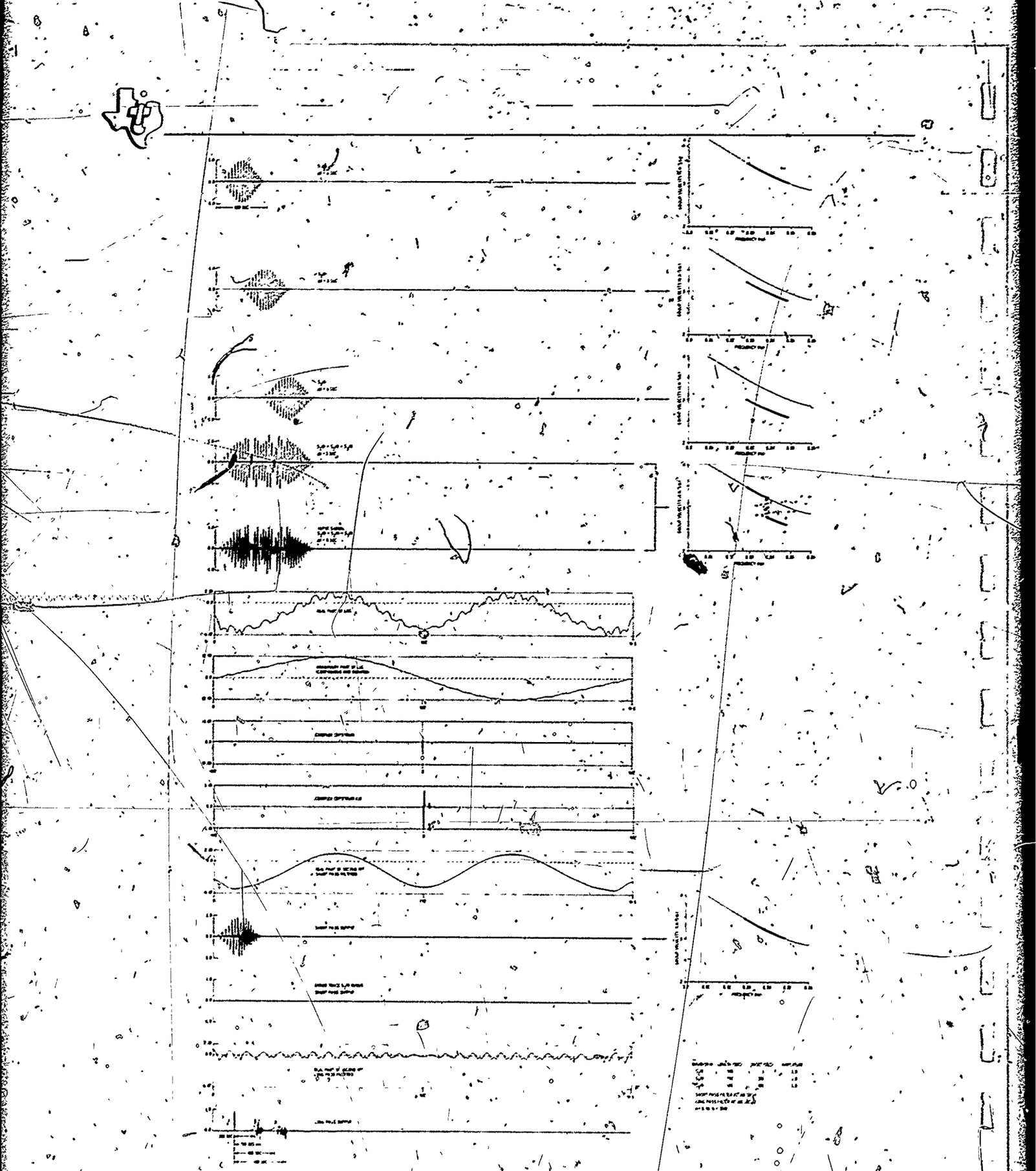


Figure V-6. Cepstrum Results for Synthetic Multipath Rayleigh Waves. Input Is the Sum of Three Chirp Waveforms; 500 Sec, 550 Sec and 600 Sec in Length with Onsets at 1 Sec, 201 Sec and 401 Sec and Amplitudes of 1 Unit, 1 Unit and 1 Unit.

the time delay between $S_1(t)$ and $S_2(t)$ is 200 sec at 0.025 Hz and increases with increasing frequency to 250 sec at 0.050 Hz while the time delay between $S_1(t)$ and $S_3(t)$ is 400 sec at 0.025 Hz and increases with rising frequency to 500 sec at 0.050 Hz.

The fifth multichirp waveform analyzed is identical to the first example with the following exception. The waveform $S_2(t)$ is scaled to one-half the amplitude of $S_1(t)$. This is clearly reflected in the long-pass output for this example (Figure V-7) where m(t) is essentially $\delta(t) + 0.5\,\delta(t-249)$.

The sixth multichirp waveform analyzed consists of two chirp waveforms 500 sec in length with onsets at 1 sec and 101 sec (Figure V-8). The time separation between $S_1(t)$ and $S_2(t)$ is much less than in previous examples. In addition, $S_2(t)$ is scaled to one-half the amplitude of $S_1(t)$ and the waveform inverted with respect to $S_1(t)$. The complex cepstrum is long-pass filtered at 80 sec, yielding m(t) which is essentially $\delta(t) - 0.5\,\delta(t-100)$. The complex cepstrum is also short-pass filtered at 80 sec, yielding the output shown. The calculated error for this output, although still fairly small, is larger than obtained from the previous examples. The short-pass system has removed all of the contribution caused by m and part of that caused by S.

The seventh multichirp waveform analyzed consists of two chirp waveforms 400 sec and 600 sec in length with onsets at 1 sec and 1 sec. The analysis results are presented in Figure V-9. For this example the time delay between $S_1(t)$ and $S_2(t)$ is zero at 0.025 Hz and increases with increasing frequency to 200 sec at 0.050 Hz. An inspection of the complex cepstrum shows that S and m overlap significantly in the "short time" region, consequently the portion attributed to m cannot be removed without removing part of the contribution due to S and vice versa. A suite of short-pass filters at 8, 16, 24, 40, 56, 72, and 88 sec was applied to the complex cepstrum. The filter at 56 sec yielded the smallest error trace. The short- and long-pass filtered results for that filter are shown in the figure. Simple short- or long-pass filtering cannot completely separate S from m. Comb filtering could be used to remove the part attributed

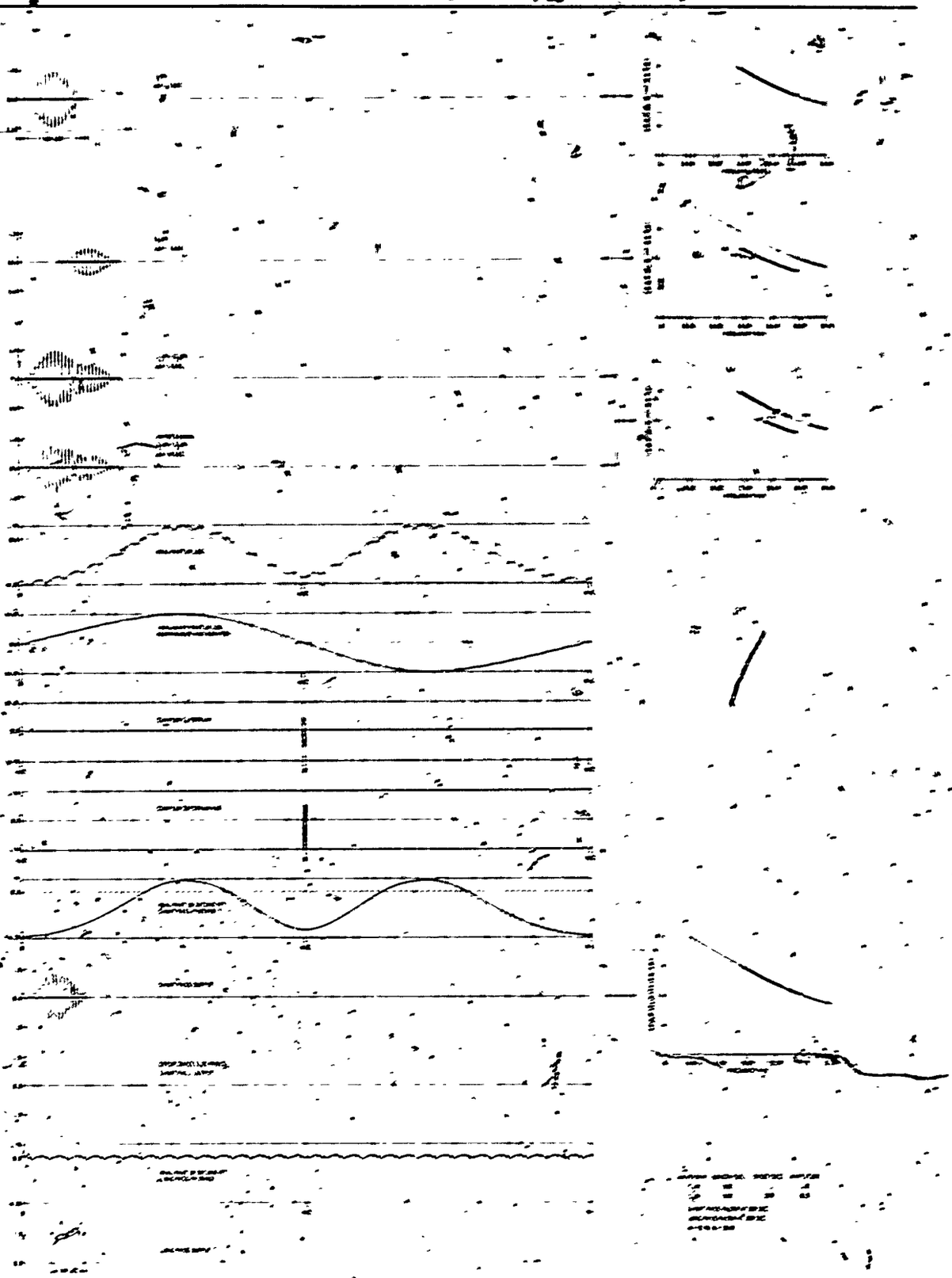


Figure V-7. Cepstrum Results for Synthetic Multipath Rayleigh Waves. Input Is the Sum of Two Chirp Waveforms; 500 Sec and 500 Sec in Length with Onsets at 1 Sec and 250 Sec and Amplitudes of 1 Unit and 1/2 Unit.

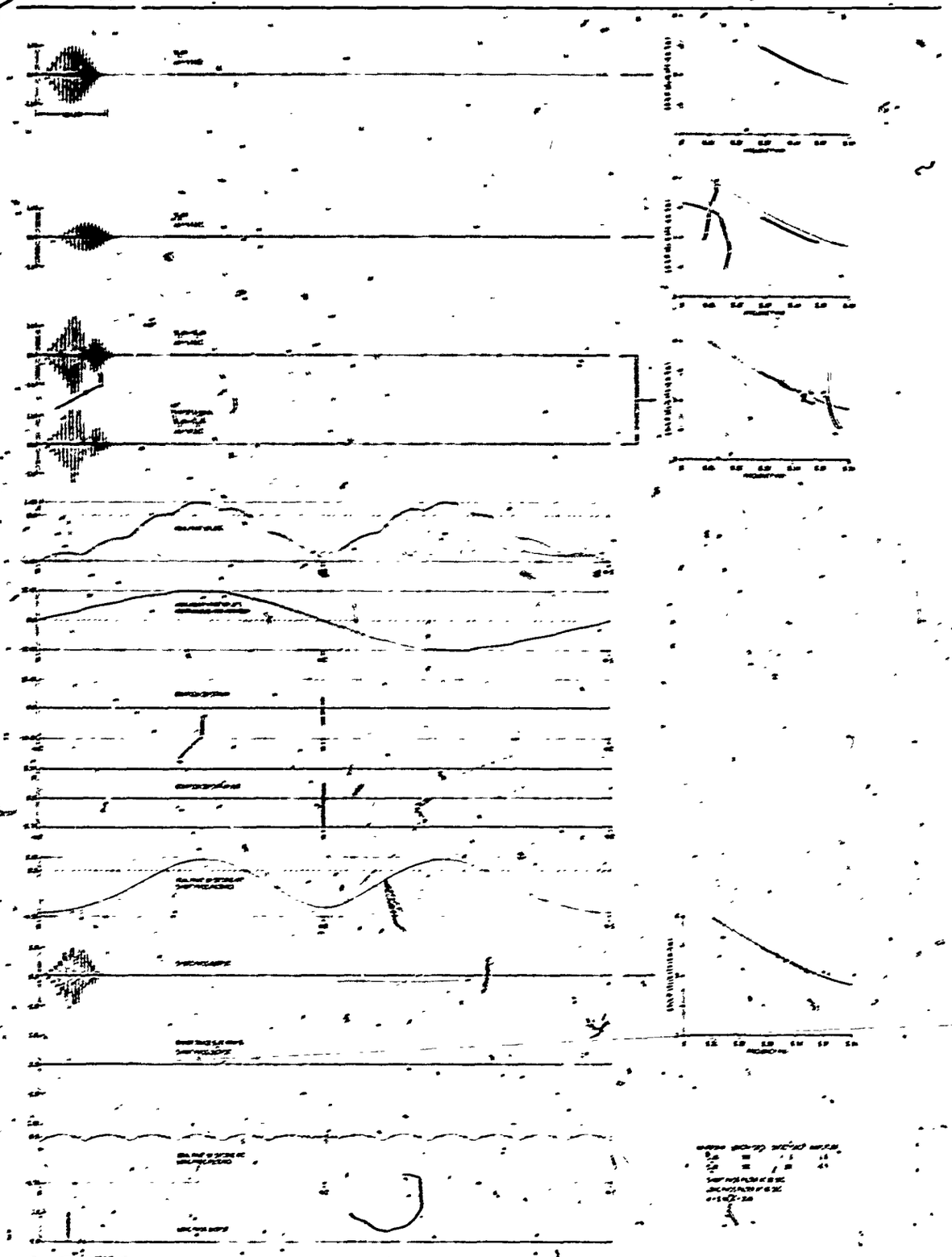


Figure V-8. Cepstrum Results for Synthetic Multipath Rayleigh Waves. Input Is the Sum of Two Chirp Waveforms; 500 Sec and 500 Sec in Length with Onsets at 1 Sec and 101 Sec and Amplitudes of 1 Unit and 1/2 Unit. (The Second Chirp Waveform, with Onset at 101 Sec, is Inverted with Respect to the First Chirp Waveform.)

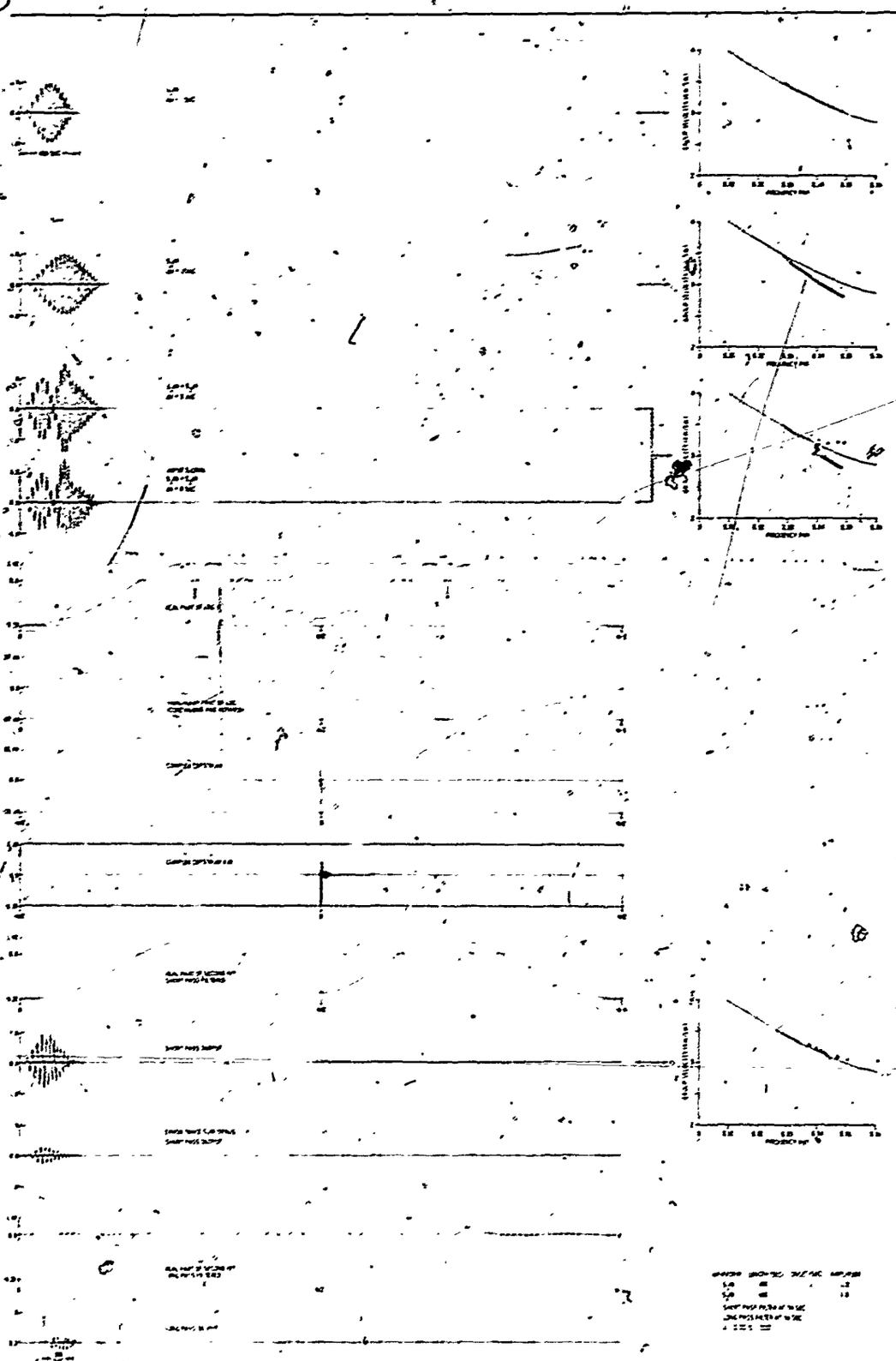


Figure V-9. Cepstrum Results for Synthetic Multipath Rayleigh Waves. Input Is the Sum of Two Chirp Waveforms; 400 Sec and 600 Sec in Length with Onsets at 1 Sec and 1 Sec and Amplitudes of 1 Unit and 1 Unit

to m while retaining almost all of the part from S, provided m was a fairly simple impulse train. However, for this example, m represents a transfer function of considerable length with zero time delay at the lowest frequency (0.025 Hz). The part attributed to m in the complex cepstrum is a complicated series of overlapping transfer functions occupying the "short time" region to which comb filtering cannot be successfully applied. When the two components significantly overlap, then the linear filter system removes too much of the complex cepstrum of the desired output. However, for this example, the signal estimate obtained is clearly a better approximation of $S_1(t)$ than is the contaminated input signal and discriminants (AR, RMS, chirp statistic, etc.) calculated from the signal estimate will more nearly approximate the true values. In addition, it should be pointed out that the group velocity points obtained from the short-pass output (Figure V-9) closely fit the expected group velocity curve of $S_1(t)$.

The results presented in this section demonstrate the feasibility of this technique for fairly simple multisource and/or multipath waveforms. The multichirp inputs tested represent simulations of teleseismically recorded Rayleigh waves resulting from:

- Interference from two or more seismic sources occurring at the same location with small time separation and no multipath propagation
- Interference from sources at different locations which arrive at the recording site with small time separation
- Interference from a single source generated Rayleigh wave which propagates along two or three distinct paths with differing group velocity characteristics therefore arriving at the recording site with small time separations

The examples show that for cases where S and m are separated in

"time", a linear filter system can be used effectively to separate the two components from one another. The technique requires no a priori estimate of S and will give an estimate of both the signal S and the multipath operator m.

SECTION VI

PRELIMINARY CEPSTRUM ANALYSIS OF RAYLEIGH WAVES RECORDED AT LASA AND ALPA

Several events recorded at LASA and ALPA were processed using the complex cepstrum technique in order to make a preliminary evaluation of the method as applied to actual Rayleigh wave recordings. The events used had good signal-to-noise ratio and associated information for the events (location, date, origin time, etc.) is given in Table VI-1. The vertical component of the Rayleigh wave was time partitioned from each event recording for input to the program.

Processing results for event NTS03 recorded at ALPA are presented in Figure VI-1. Shown in the figure are the decimated input signal, the real and imaginary parts of the log, the complex-cepstrum, the short-pass filtered output, and the long-pass filtered output. The input signal has a modulated character, and minima are observed in the plot of the real part of the log. These minima could of course result from either an interference phenomenon or low signal excitation at the source at those frequencies. Whether or not energy at those frequencies was present at the source cannot be ascertained from this beamed output alone, however an inspection of the group velocity curve calculated from this event (Figure VI-2, top) shows that the recorded waveform is not a simple normally dispersed Rayleigh wave and this characteristic is indicative of interfering waveforms.

The complex cepstrum for this event is much more complicated than those obtained for the simple examples given in the previous section. The part of the complex cepstrum attributed to m is not a simple impulse train. Thus it would be very difficult to apply a comb filter system since this requires much information about the number, locations and durations of the impulses in the complex cepstrum. An alternative is to use a short-pass system. A suite of short-and long-pass filters covering the interval from ± 16 sec to ± 128 sec

Table VI-1

EVENT INFORMATION CATALOGUE

Event	Event Location	Lat.	Long.	Date	Origin Time	Mag.	Depth (Km)	Dist. (Deg.)	Azim. (Deg.)	Record. Site
L - 01	Greenland Sea	73.4N	6.8E	18 Nov 1966	18:48:44	4.9	33	51.9	19.7	LASA
NTS03	Southern Nevada	37.3N	116.5W	26 Mar 1970	19:00:00	6.5	0	33.4	131.4	ALPA
EPX 14649	Southern Norway	63.6N	10.0E	29 Sep 1969	10:27:39	4.5	15	59.1	27.8	LASA
EPX 14646	S. Sinkiang Province	41.5N	89.5E	29 Sep 1969	8:40:31	4.7	33	91.1	348.3	LASA
ALPA 1001	S. Sinkiang Province	39.9N	77.8E	29 Jul 1970	5:50:56	5.2	13	69.4	324.1	ALPA
LL-2008	Kurile Is.	46.7N	152.5E	21 Nov 1966	12:19:27	5.6	40	64.3	311.5	LASA
LL-2018	Kurile Is.	46.3N	152.9E	1 Apr 1967	5:57:09	5.5	40	64.0	311.0	LASA
LL-2019	Kurile Is.	45.7N	151.8E	1 Apr 1967	12:23:36	5.9	40	65.3	311.0	LASA
LL-2020	Kurile Is.	45.8N	151.7E	1 Apr 1967	14:00:34	5.4	23	65.3	311.2	LASA
LL-2027	Kurile Is.	47.5N	155.4E	7 Jun 1967	18:16:31	5.2	29	62.3	310.7	LASA
LL-2028	Kurile Is.	45.4N	150.7E	5 Oct 1967	15:55:03	5.3	33	66.1	311.4	LASA

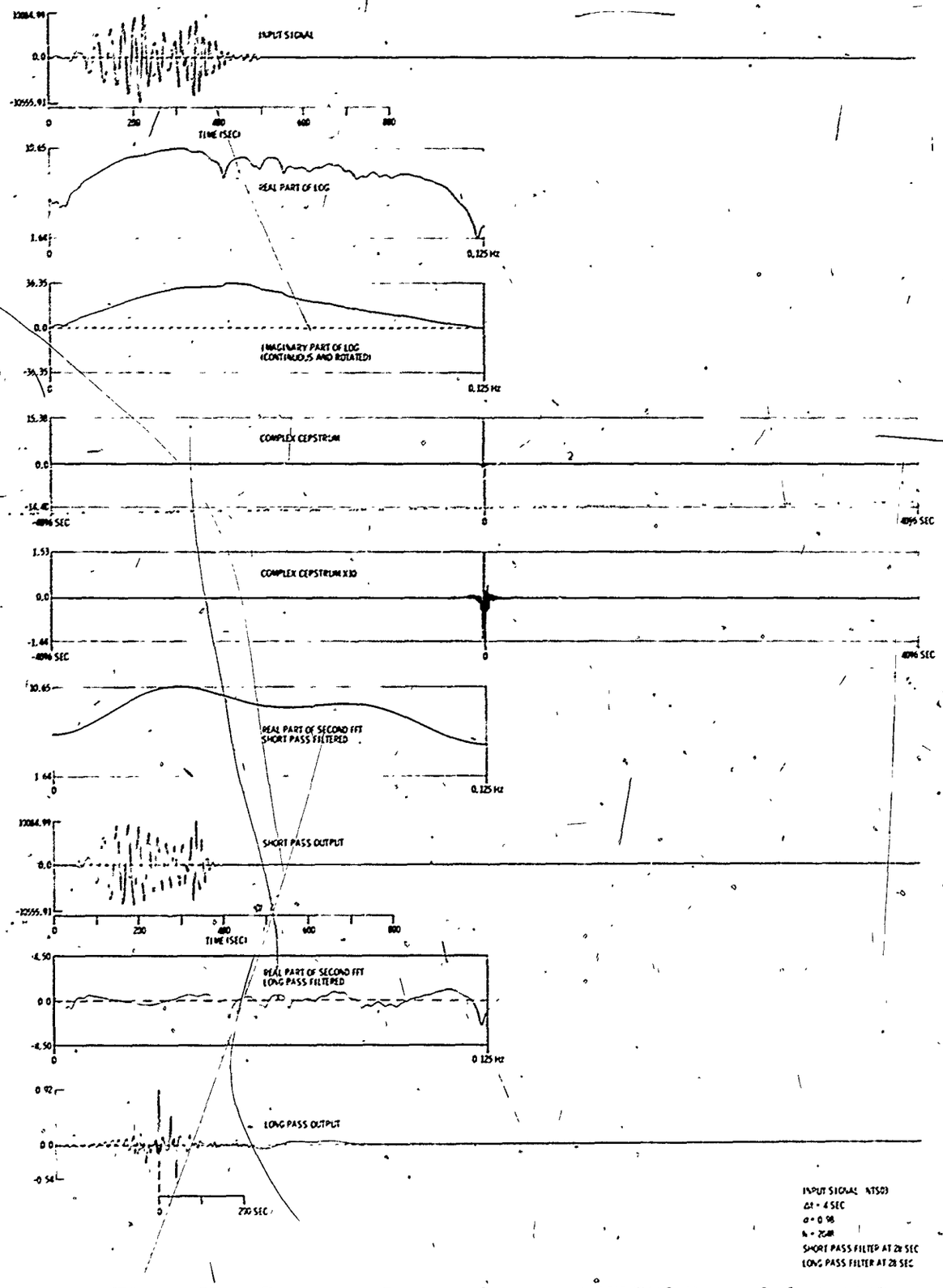


Figure VI. Cepstrum Results for Event NTSO3 Recorded at ALPA

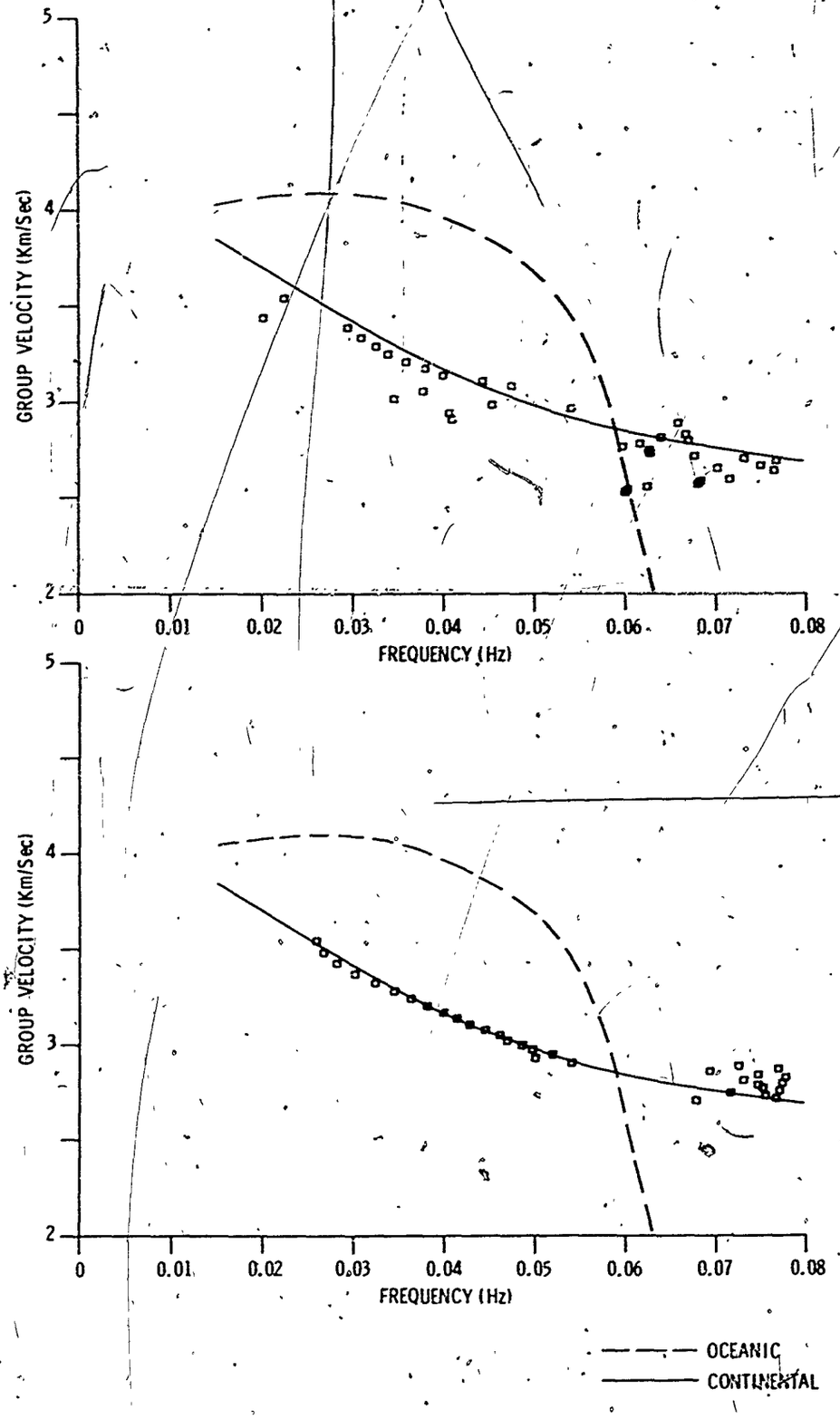


Figure VI-2. Group Velocity vs Frequency for NTSQ3. (Top, Input Signal; Bottom, Short-Pass Output)

was applied to the complex cepstrum and outputs obtained. An analysis of those results was made by inspection of the plots showing the real part of the second FFT after short-and long-pass filtering. The short-pass filters at 16 and 20 sec removed all of the rapidly varying component due to m and also removed much of the part due to S. The filters at 36 to 128 sec passed almost all of the component due to S and also passed increasing amounts of the rapidly varying component due to m for increasing filter times. The filters at 24 to 32 sec passed most of the part due to S and removed almost all of the rapidly varying component. Shown in the figure are the real part of the second FFT and the output for both short-and long-pass filters at 28 sec.

Group velocity was calculated from the short-pass output. A comparison between the group velocity points obtained from the input waveform and the short-pass output (Figure VI-2, top and bottom respectively) shows that the scatter of points is largely reduced for the output trace and the points are much more continuous. In addition, the points closely fit the theoretical curve calculated for a continental crust. The great circle path for this event traverses Nevada, Oregon, Washington, parallels the Coast Mountains of British Columbia, and crosses a portion of Alaska.

The long-pass output shows a large complex impulse at about 30 sec. An interpretation of the long-pass output based on several assumptions could of course be made. However, additional information would be required to substantiate that the assumed conditions and therefore the interpretation were correct. For example, two sources with identical excitation and location occurring with small time separation would yield a long-pass output as shown for examples one and two of the previous section. However, it is conceivable that an entirely different physical situation could yield an essentially identical waveform recording and subsequent long-pass output. Therefore an analysis of short-period p waves recorded at a network of stations and subsequent hypocenter calculations would be necessary in order to confirm or deny the possibility that two sources of the same type and magnitude occurred at the same location

with the time separation observed in the long-pass output. Conversely, knowledge of the information described above would allow accurate filter placement in the complex-cepstrum technique to yield the optimum signal estimate.

Next, a Greenland Sea event L-01 recorded at LASA was processed. Those results are presented in Figure VI-3. The great circle path for this event traverses a portion of the Greenland Sea, Greenland and the Canadian Shield. Group velocities calculated from the input signal and the short-pass output are shown in Figure VI-4. The input signal is relatively free of beats and is a fairly simple normally dispersed waveform as seen in the group velocity plot. However, interference is occurring late in the waveform and a large scatter is present at the low group velocities which correspond to the coda of the event. Group velocity for the short-pass output is somewhat smoother, and as seen in Figure VI-3, much of the coda of the input event is not present in the signal estimate. The long-pass output shows a large complex impulse at about 60 sec. The long ringing train around 300 to 600 sec is attributed to the most rapidly varying component, seen in the plot of the real part of the log, with a Δf of about 0.002 to 0.003 Hz.

Four events from the Kurile Is. recorded at LASA were processed. Rayleigh waves recorded at LASA from events in this region are notoriously complex, and parameters calculated from the waveforms give results which are difficult to interpret as evidenced by the group velocity plots shown in Section V, Figure V-3. The great circle path traverses the Kurile Arc, Bering Sea, Gulf of Alaska, and Western Canada — a complex series of structural environments. Complete cepstrum processing results for LL-2008 and LL-2018 are shown in Figures VI-5 through VI-8. For brevity, only the input signals, short-and long-pass outputs and calculated group velocities are shown for events LL-2019 and LL-2028 (Figure VI-9). All four events show beats in the input signal and minima are observed in the plots showing the real part of the log. Again the complex cepstra are not simple impulse trains and are difficult to interpret. A suite of short-pass filters was applied to the complex cepstra and outputs calculated as before. Presented in the figures are the outputs for a filter at

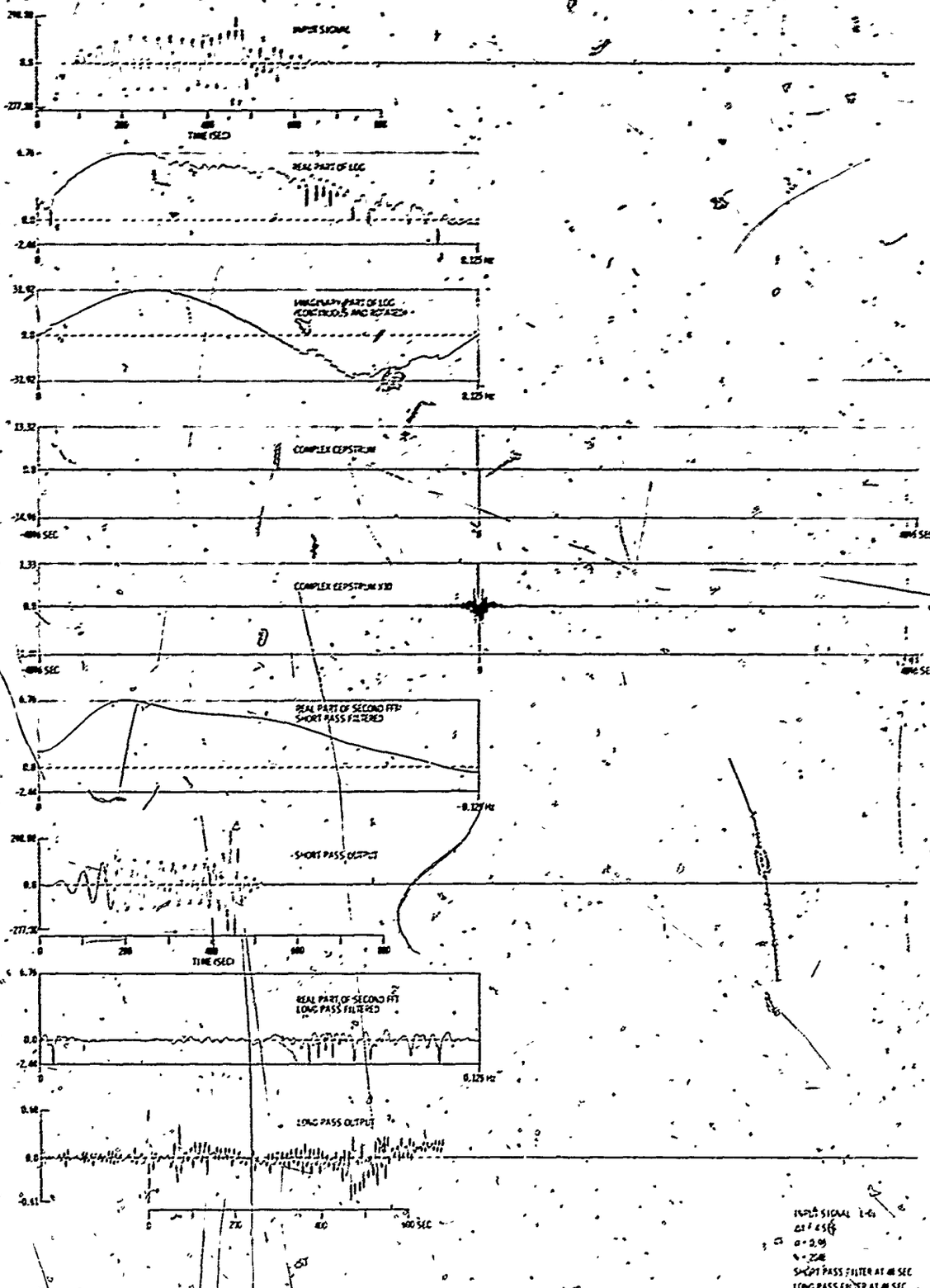
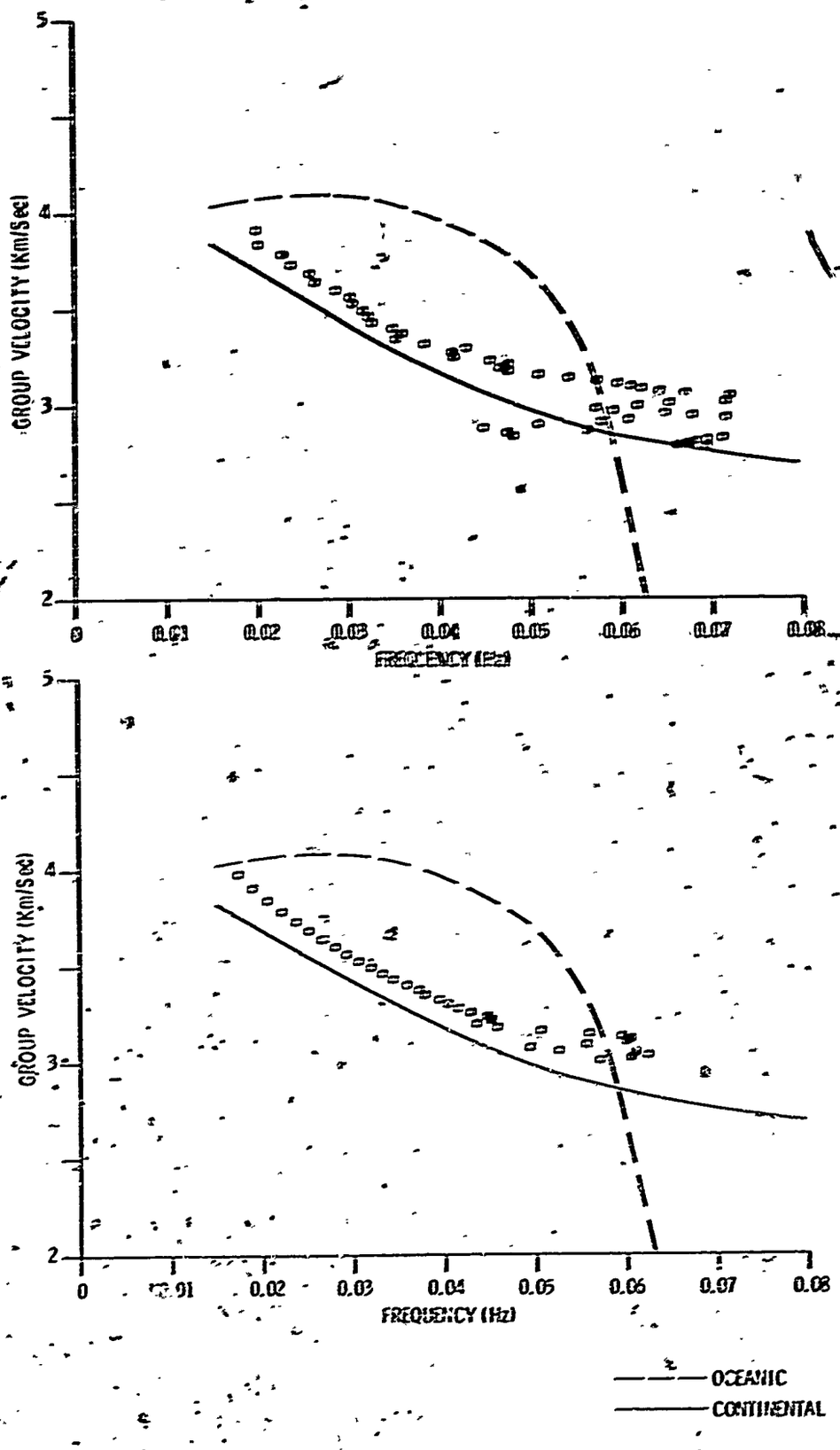


Figure VI-3. Cepstrum Results for Event L-01 Recorded at LASA



igure IV-4. Group Velocity vs Frequency for L-01. (Top, Input Signal; Bottom, Short-Pass Output)

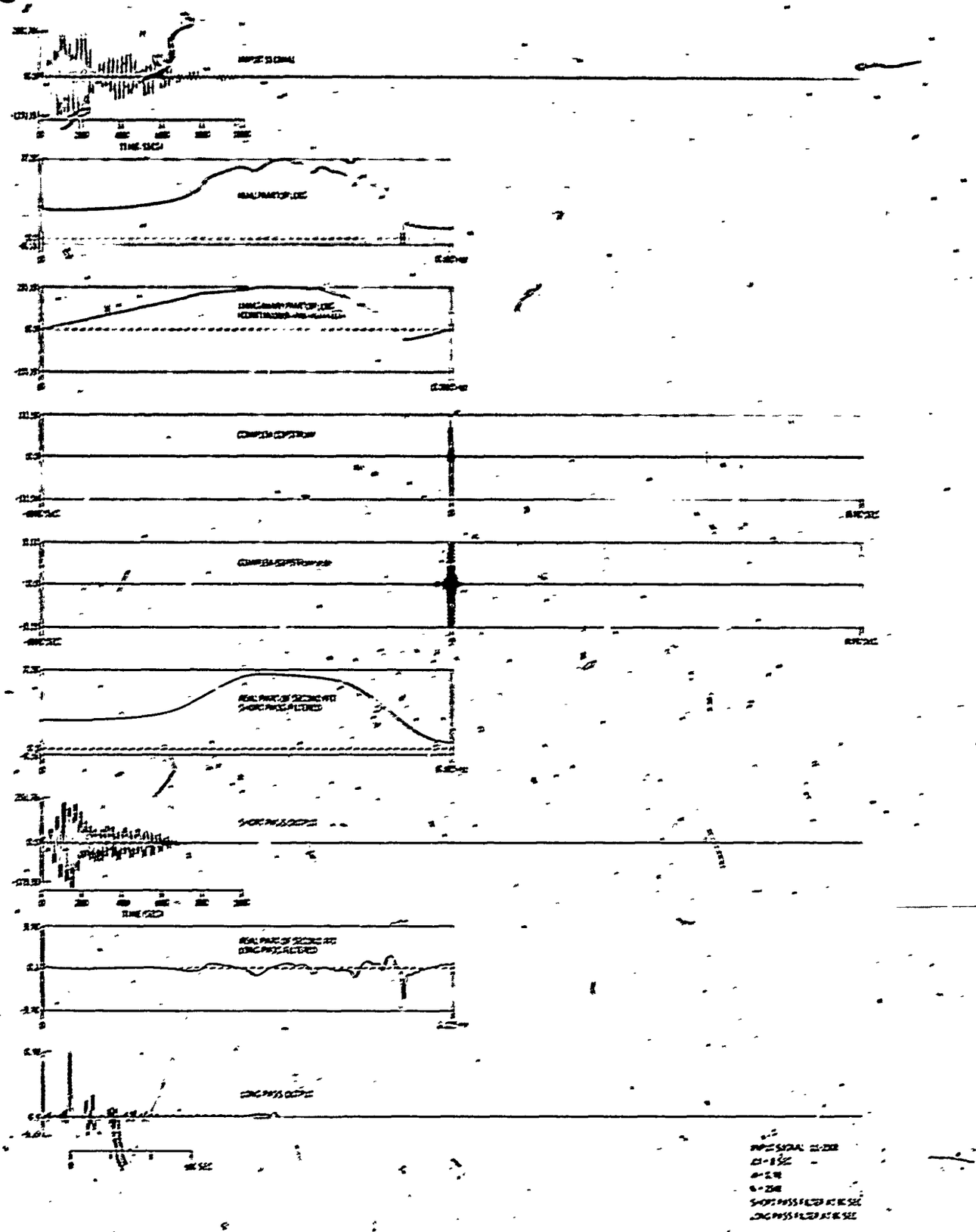


Figure VI-5. Cepstrum Results for Event LL-2008 Recorded at LASA

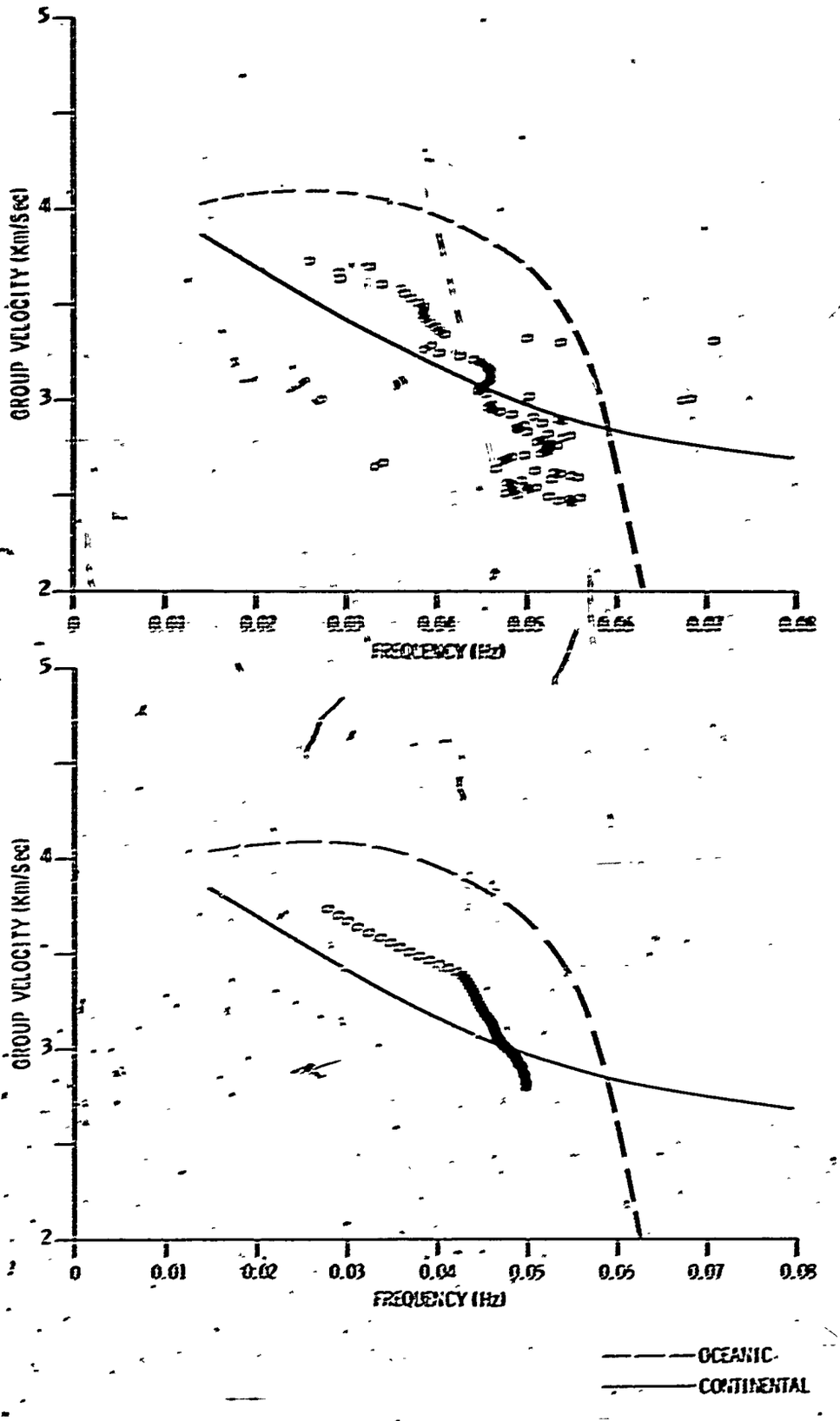


Figure VI-6. Group Velocity vs Frequency for LL-2008. (Top, Input Signal; Bottom, Short-Pass Output)

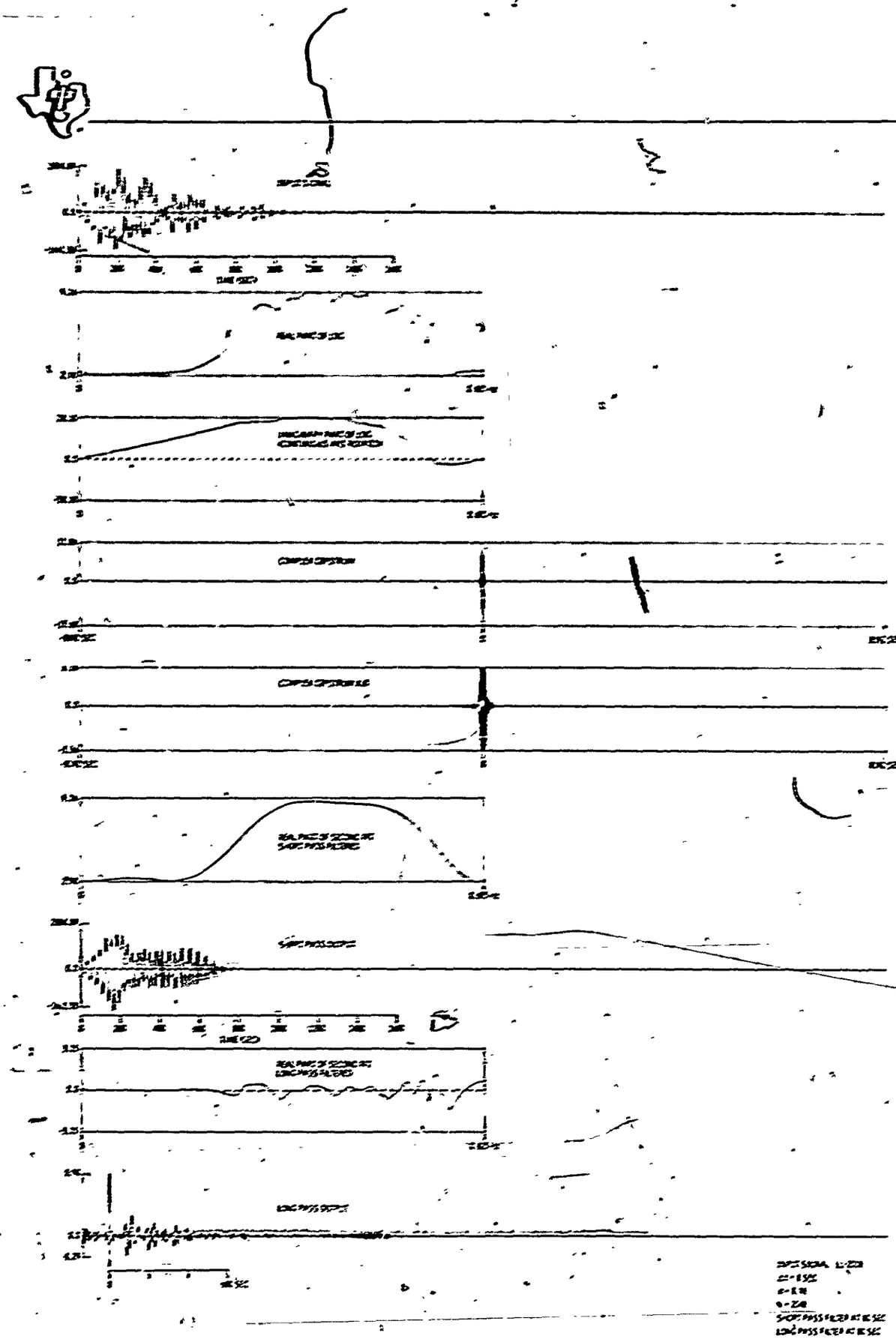


Figure VI-7. Cepstrum Results for Event LL-2018 Recorded at LASA

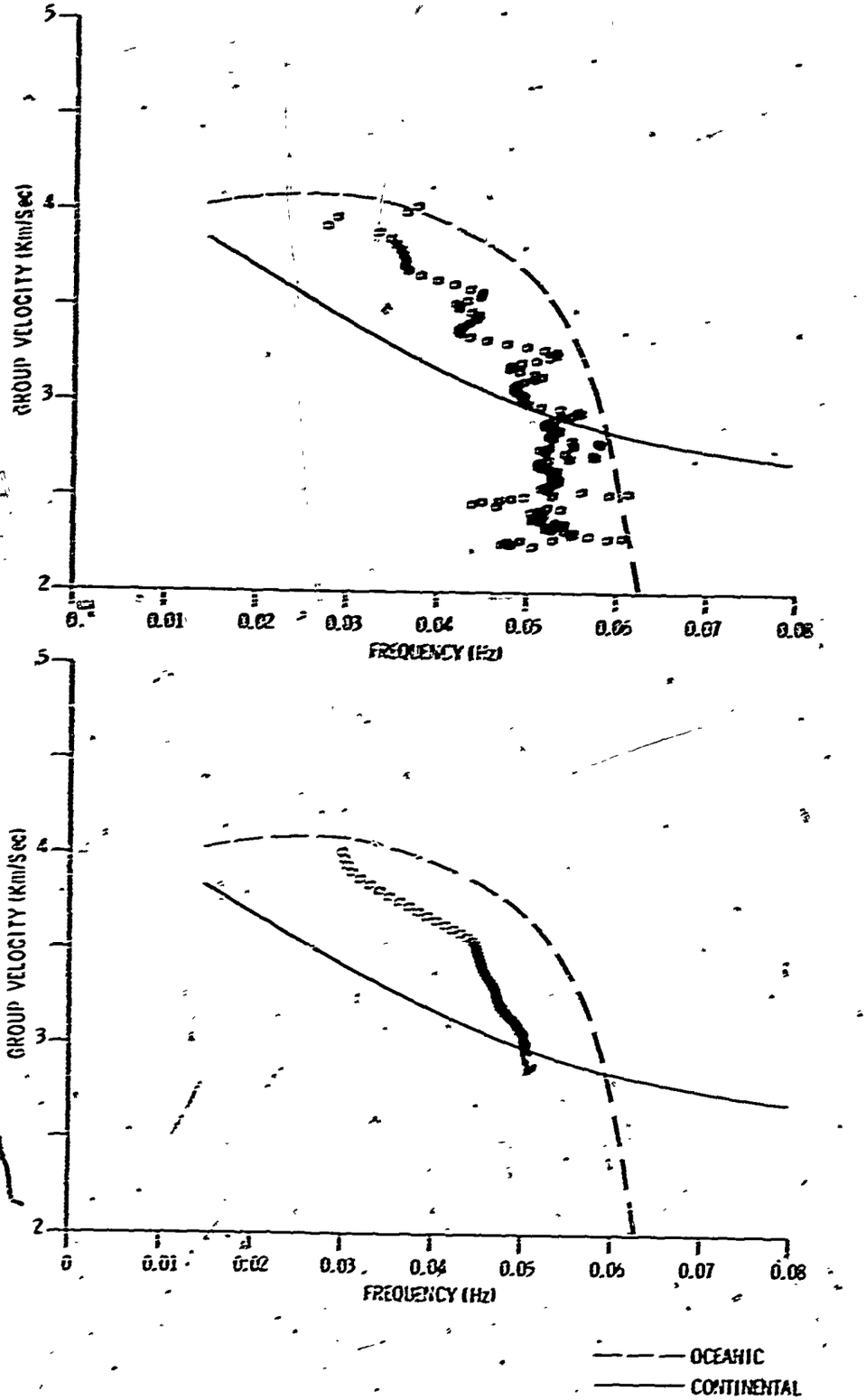


Figure VI-8. Group Velocity vs Frequency for LL-2018. (Top, Input Signal; Bottom, Short-Pass Output)

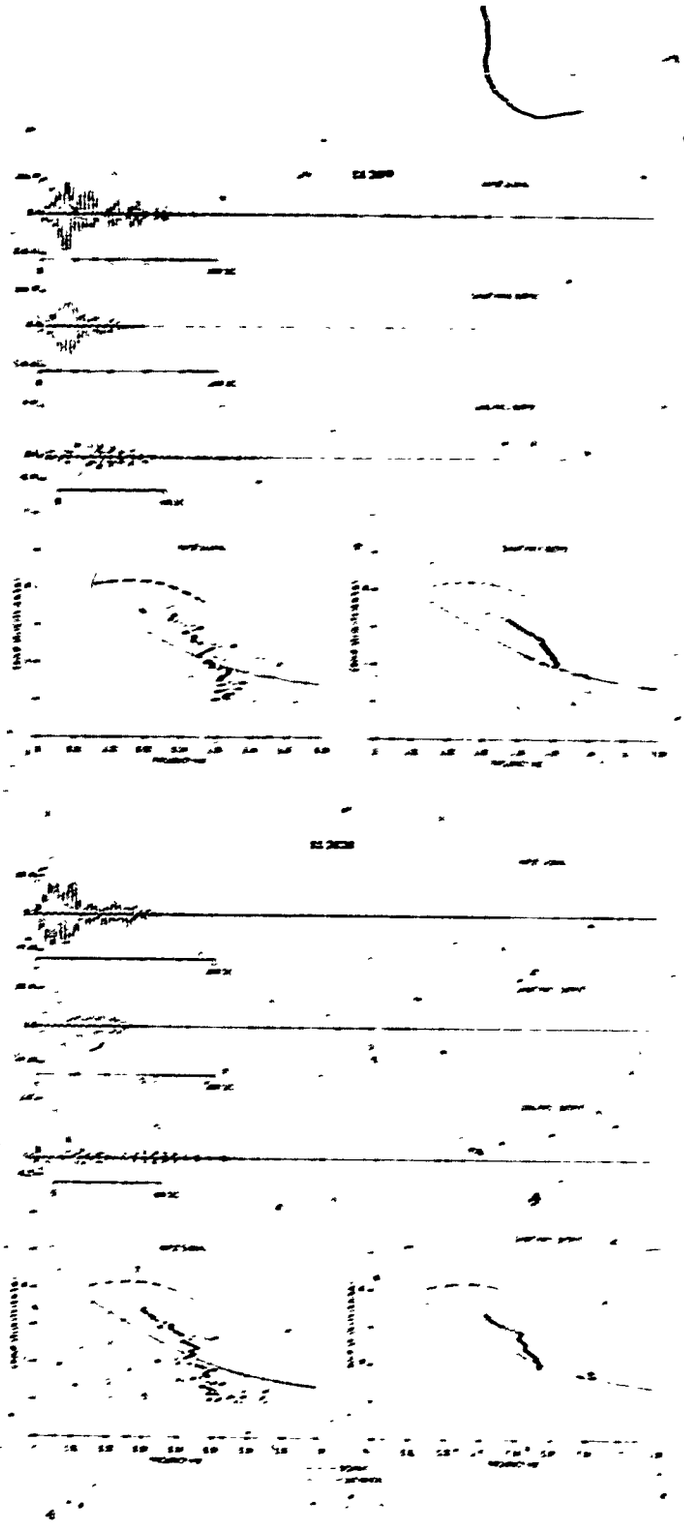


Figure VI-9. Events LL2019 and LL-2028. (Input Signals, Cepstrum Outputs, and Group Velocities)

80 sec. The signal estimates obtained are clearly much simplier than the input waveforms. The LL event recordings obtained from Lincoln Laboratories were bandpass filtered at 0.025-0.055 Hz during preprocessing. Since there is no energy above 0.0625 Hz, the event recordings were decimated by 8 giving a sample period of 8 sec and a Nyquist frequency of 0.0625 Hz at input to the complex cepstrum program. Group velocity points calculated from the input waveform sampled at 8 sec showed variations with respect to the waveform sampled at 1 sec. Those variations occurred only at the higher frequencies where frequency is changing very slowly with time for these events. The three-point quadratic interpolation used in the group velocity calculations to determine peak and trough times is adequate over the frequency band of interest for waveforms sampled as coarse as 4 sec and no smoothing was performed on those waveforms with $\Delta t \leq 4$ sec; however for the waveforms sampled at 8 sec smoothing of the order number vs time function over about 5 points was necessary to reduce the variation in calculated group velocity at the higher frequencies. It should be pointed out that smoothing over 5 points does not significantly alter the group velocity points calculated from the original waveform sampled at 1 sec.

Group velocities calculated from the input signals and the short-pass filtered outputs are presented in Figures VI-6, VI-8 and VI-9. The simplicity of the group velocity curves for the outputs is apparent. The curves obtained are neither oceanic nor continental but are an intermediate variant of the two. Group velocity calculations are based on the epicentral distance between event and recording site which assumes a great circle path for the surface waves. If the first arriving surface waves (at a given frequency) have deviated from the great circle path by traversing a higher velocity structure, then the calculated group velocities will be lower than for the actual path taken.

The events were long-pass filtered at 80 sec, yielding a fairly simple output for several of the events wherein 4 or 5 separate complex impulse trains may be observed. The similarity of the outputs obtained is strongly suggestive of several fairly distinct propagation paths from source to receiver

for these events. This type of analysis coupled with frequency-wavenumber analysis (Capon,1970) could provide a powerful tool for understanding the complex nature of recorded Rayleigh waves.

Six additional event recordings were processed using the complex-cepstrum technique. The results of these calculations are presented simply by showing the group velocities calculated from the input signals and those calculated from the output signals (Figures VI-10, VI-11). The results obtained are similar to previous results in that the outputs are simple normally dispersed wavetrains as evidenced in the group velocity plots.

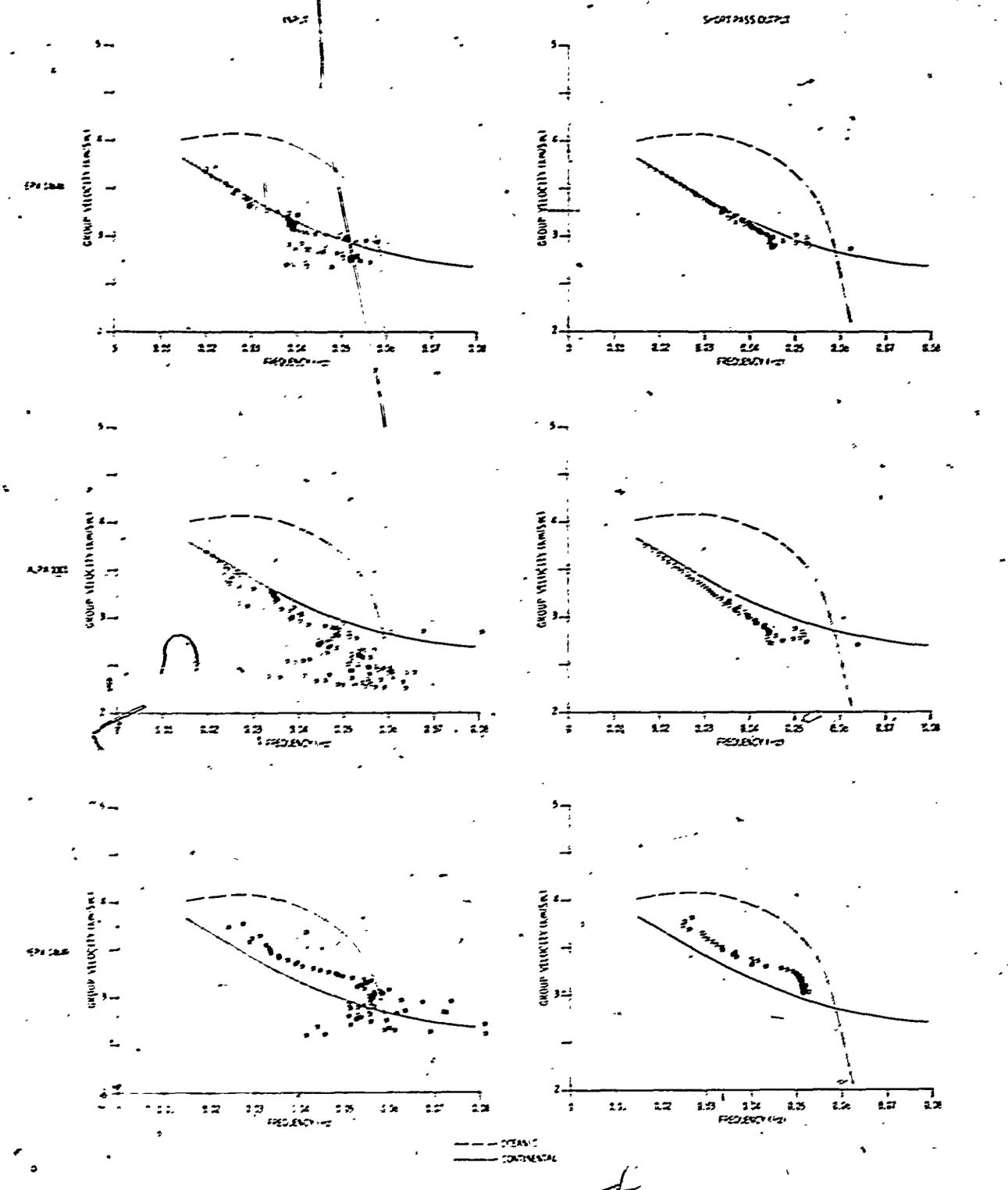


Figure VI-10. Group Velocities for Input Signals and Short-Pass Outputs. Filter at 64 sec. Events EPX 14646, ALPA 1001, and EPX 14649

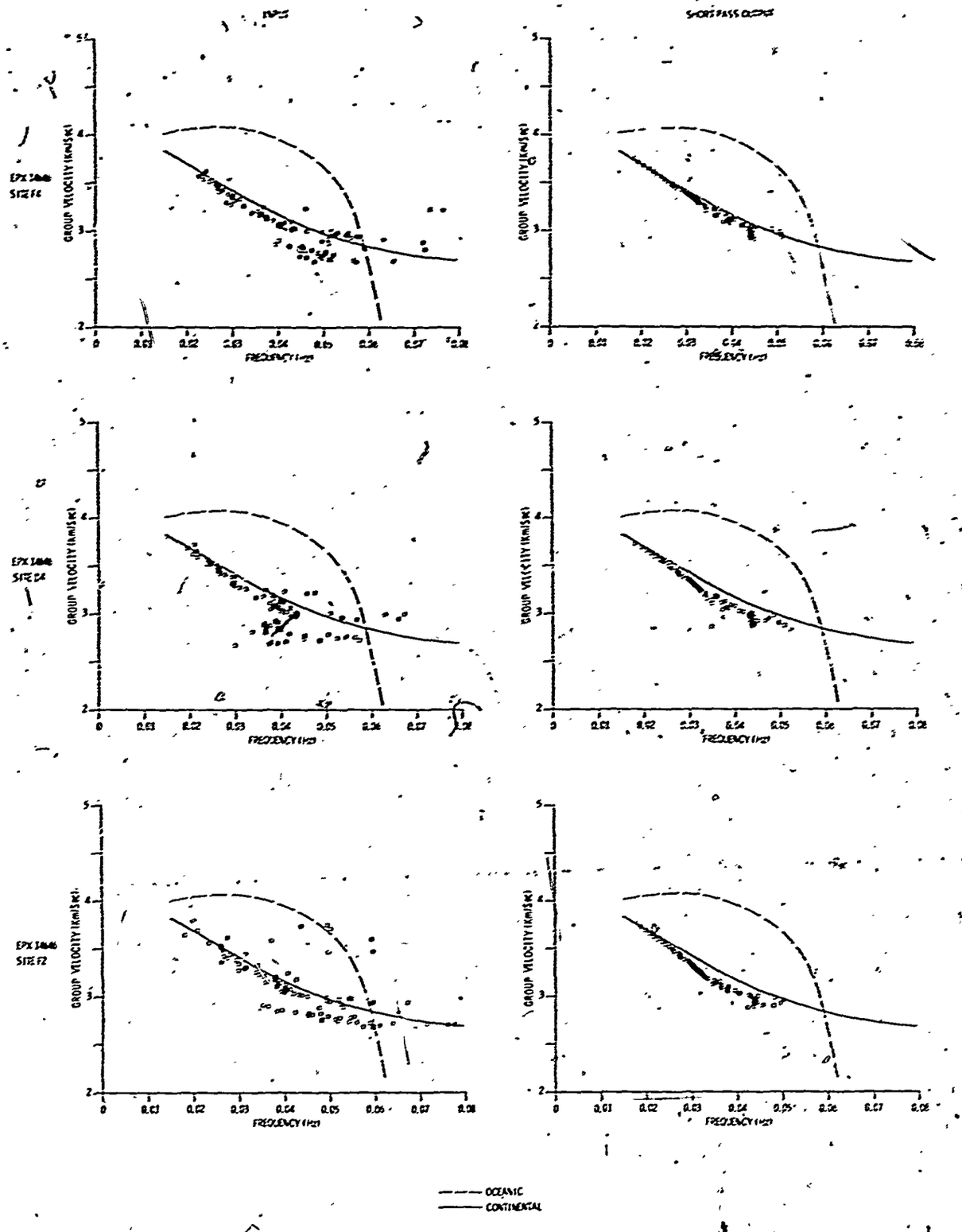


Figure VI-11. Group Velocities for Input Signal and Short-Pass Outputs. Filter at 64 sec. Input Signals are Single-Channel Recordings of Event EPX 14646

SECTION VII

REFERENCES

Capon, Jack, 1970, Analysis of Rayleigh-wave multipath propagation at LASA: Bull. Seism. Soc. Am., 60, p. 1701-1731.

Evernden, J.F., 1953, Direction of approach of Rayleigh waves and related problems, Part I: Bull. Seism. Soc. Am., 43, p. 335-374.

Evernden, J.F., 1954, Direction of approach of Rayleigh waves and related problems, Part II: Bull. Seism. Soc. Am., 44, p. 159-184.

Knopoff, L., S. Mueller, and W.L. Pilant, 1966, Structure of the crust and upper mantle in the Alps from the phase velocity of Rayleigh waves: Bull. Seism. Soc. Am., 56, p. 1009-1044.

Pilant, W., L. Knopoff, 1964, Observations of multiple seismic events: Bull. Seism. Soc. Am., 54, p. 19-39.

Schafer, Ronald W., 1969, Echo removal by discrete generalized linear filtering: Technical Report 466, Massachusetts Institute of Technology, Research Laboratory of Electronics, Cambridge, Mass., Ph.D. Thesis.

Texas Instruments Incorporated, 1967, Continuation of basic research in crustal studies: Final Report. Contract AF 49 (638-1588.

APPENDIX A

COMPLEX CEPSTRUM PROGRAM DOCUMENTATION

I DESCRIPTION

Homomorphic deconvolution using a generalized concept of linear filtering is applied to separating the components of a convolution. The method is based on transforming a convolution of waveforms into a sum, using a linear filter system to separate the additive components and transforming the result back to the original input space. An extensive presentation of the approach is given by Ronald W. Schafer (1969), Echo Removal by Discrete Generalized Linear Filtering: Technical Report 466, Massachusetts Institute of Technology, Cambridge, Mass., PhD Thesis.

A typical run time for the complex cepstrum program required approximately 3 minutes on the 360/65.

CalComp plots of the inputs and outputs, as well as various functio obtained during the sequence of calculations are generated to allow visual analysis of the results.

II RESTRICTIONS

1) The input data trace may not exceed 1946 points.

2) The program is dimensioned for a maximum of 2048 complex numbers in the FFT calculations.

III INPUT

Card 1 NN100

Card 2

Column	Variable	Mode	Description
1-10	NKEPS	I	Number of events to process in this run.

Card 3

1-60	NAME	A	Alphanumeric title

Card 4

1-10	IDIN	I	Input Trace ID
11-15	NSKIP	I	Number of points to skip on input
16-20	NPTS	I	Number of points to read on input

Card 4 (continued)

Column	Variable	Mode	Description
21-25	INL	I	Decimation interval after band limiting
26-30	IFILT	I	=0 stop after calculating the complex cepstrum. >0 number of filters to apply
31-35	IAARON	I	=1 call Aaron's algorithm =2 call Schafer's algorithm (for correcting phase)
36-40	NLEN	I	Total data length for transforms, etc. (must be a power of two). Zeros are added to give an input trace NLEN points in length.
41-50	DBLIM	E	>0 clip real part of log at DBLIM db below peak. <0 no clipping.
51-60	ALPHA	E	= α. Exponentially weight the input trace X. $W(n) = \alpha^n X(n)$.
61-70	DT	E	Sample period of the input data.

Card 5

1-10	IDOUT	I	Output ID for data to be saved on tape.
11-20	IDIS	I	Output disposition for data to be saved on tape.

Card 6 Filter cards, Supply IFILT times

1-10	ITYPE	I	=1 short-pass system. =2 long-pass system. =3 comb filter
11-20	INITPT	I	Point at which to filter the complex cepstrum (if ITYPE =1 or 2). Number of comb filter points (if ITYPE=3).
21-30	IDX	I	2*IDX+1 points will be zeroed at each comb location (supply if ITYPE=3).
31-40	JTAPER	I	=0 cosine taper the long-or short-pass filter. =1 no taper on filter.

Card 7 Comb Filter cards, (Format 8I10) supply if ITYPE=3).

Supply (INOTCH(J), J=1, INITPT) comb centers.

Column	Variable	Mode	Description
Card 8	NN100		
I/O	units needed		
	5 - card reader		
	6 - printer		
	7 - card punch		
	Input data tape.		
	Output tape to save results.		
	CalComp plot tape.		

```
C     COMPLEX CEPSTRUM PROGRAM
C     TI<>FFT<>LOG<>FFT<>PT<>IFFT<>FILTER<>FFT<>RPT<>EXP<>IFFT<>IT
C     THIS IS A TECHNIQUE DEVELOPMENT VERSION WHICH CAN BE SIMPLIFIED
C     APPRECIABLY AS WELL AS SPEEDED UP FOR PRODUCTION TYPE RUNS
C     DIMENSIONED FOR 2048 REAL DATA POINTS
      DIMENSION AREA(8000),NAME(15),AUM(7)
      DIMENSION X(2048),WORK(2049)
      DIMENSION TEMP(2048)
      DIMENSION COR(2048)
      DIMENSION SAVE(2048)
      DIMENSION TRACE(2048)
      DIMENSION SIGR(2048)
      DIMENSION NOTCH(100)
      DIMENSION TAPER(8)
      COMPLEX X,SAVE
      DATA TAPER/.9848,.9397,.8660,.7660,.6424,.5000,.3420,.1736/
      CALL NN100
   10 FORMAT(1X,10E15.7)
      PI=3.141593
      TWOPI=6.283185
      CALL PLOTS(AREA,8000,3.0)
      CALL DAY(IDATE)
      READ(5,7) NKEPS
      DO 1000 IKEPS=1,NKEPS
      CALL STASK
      READ(5,1) NAME
    1 FORMAT(15A4)
      WRITE(6,2) NAME
    2 FORMAT(1H1,1X,'NAME = ',15A4)
      WRITE(6,3) IDATE
    3 FORMAT(//,1X,'DATE =',I10)
      READ(5,4) IDIN,NSKIP,NPTS,INC,IFILT,IAARON,NLEN,DBLIM,ALPHA,DT
    4 FORMAT(I10,6I5,3F10.0)
      WRITE(6,5)
    5 FORMAT(//,1X,'IDIN,NSKIP,NPTS,INC,IFILT,IAARON,NLEN,DBLIM,ALPHA,DT'
     1')
      WRITE(6,6) IDIN,NSKIP,NPTS,INC,IFILT,IAARON,NLEN,DBLIM,ALPHA,DT
    6 FORMAT(1X,7I10,F10.2,F10.4,F10.2)
      READ(5,7) IDOUT,IDIS
    7 FORMAT(2I10)
      WRITE(6,9) IDOUT,IDIS
    9 FORMAT(//,1X,'IDOUT = ',I10,'     ,IDIS =  ',I5)
      DELY=20.48/(NLEN*INC*DT)
C     INC=DECIMATION RATE
C     DT=SAMPLE PERIOD OF INPUT DATA
C     NPTS=NUMBER OF INPUT POINTS
      NPF=101
      F0=0.0
      FI=.025
```

```
      E2=.050
      E3=.0625
      WRITE(6,13) F0,E1,E2,E3
   13 FORMAT(//,1X,'CUTOFF FREQUENCIES FOR PREFILTER',4F10.4)
C     DESIGN ZERO PHASE BAND PASS FILTER
      FD=1./(FLOAT(NPF)*DT)
      FR=1./(2.*DT)
      NF=1+INT(FR/FD)
      FTEST=F0+F1
      F=0.
      DO 480 I=1,NF
      IF(FTEST)460,460,440
  440 IF(E0-F) 450,445,445
  445 WORK(I)=0.
      GO TO 480
  450 IF(E1-F).460,455,455
  455 WORK(I)=(1.+COS(PI*(F-E1)/(E0-E1)))/2.
      GO TO 480
  460 IF(E2-F) 470,465,465
  465 WORK(I)=1.
      GO TO 480
  470 IF(E3-F). 445,475,475
  475 WORK(I)=(1.-COS(PI*(F-E3)/(E2-E3)))/2.
  480 F=F+FD
      LF=(NPF-1)/2
      Y=-DT*FLOAT(LF)
      DO 490 I=1,NPF
      TEMP(I)=WORK(1)
      F=FD
      DO 485 J=2,NF
      TEMP(I)=TEMP(I)+2.*COS(TWOPI*F*Y)*WORK(J)
  485 F=F+FD
      Y=Y+DT
  490 TEMP(I)=TEMP(I)*FD*DT
C     READ TRACE,TRUNCATE EACH END AT NEAREST ZERO CROSSING
C     APPLY COSINE TAPER TO EACH END, BAND PASS FILTER
C     DECIMATE, REMOVE MEAN
      CALL RDCP(IDIN,WORK,NSKIP,NPTS,1,AUM)
      AMAX=0.
      SUM=0.
      DO 406 I=1,NPTS
  406 SUM=SUM+WORK(I)
      SUM=(SUM/NPTS)
      DO 407 I=1,NPTS
  407 WORK(I)=WORK(I)-SUM
      INDEX=1
      DO 815 J=1,NPTS
      IF(WORK(J).GE.0..AND.WORK(J+1).LE.0.) GO TO 816
      IF(WORK(J).LE.0..AND.WORK(J+1).GE.0.) GO TO 816
  815 INDEX=INDEX+1
      GO TO 1009
  816 IF(ABS(WORK(INDEX)).GE.ABS(WORK(INDEX+1)))INDEX=INDEX+1
      NPTS=NPTS-INDEX+1
      DO 817 J=1,NPTS
```

```
817 WORK(J)=WORK(INDEX-1+J)
    INDEX=NSKIP+INDEX-1
    WRITE(6,491) INDEX
491 FORMAT(//,1X,'TOTAL NUMBER OF POINTS SKIPPED ON INPUT =',I10)
    INDEX=NPTS
    DO 818 J=1,NPTS
    IF(WORK(INDEX).GE.0..AND.WORK(INDEX-1).LE.0.) GO TO 819
    IF(WORK(INDEX).LE.0..AND.WORK(INDEX-1).GE.0.) GO TO 819
818 INDEX=INDEX-1
    GO TO 1000
819 IF(ABS(WORK(INDEX)).GE.ABS(WORK(INDEX-1)))INDEX=INDEX-1
    NPTS=INDEX
    J=NPTS-7
    K=8
    DO 820 I=1,8
    WORK(J)=TAPER(I)*WORK(J)
    WORK(K)=TAPER(I)*WORK(K)
    J=J+1
820 K=K-1
    NHAF=NPF/2
    DO 425 I=1,NHAF
425 TRACE(I)=0.
    DO 825 J=1,NPTS
825 TRACE(NHAF+J)=WORK(J)
    DO 100 J=1,NPTS
100 WORK(J)=TRACE(NHAF+J)
    CALL WRCP(IOOUT,WORK,NPTS,1,1,IDIS,IDATE,NAME,AUM)
    IOOUT=IOOUT+1
    K=NHAF+NPTS+1
    L=K+NHAF-1
    DO 430 I=K,L
430 TRACE(I)=0.
C   FILTER INPUT TRACE
    ZERO=TEMP(NHAF+1)
    DO 431 I=1,NPTS
    I=I-1
    K=NPF+I
    SUM=0.
    DO 426 J=1,NHAF
426 SUM=SUM+TEMP(J)*(TRACE(J+L)+TRACE(K-J))
431 WORK(I)=SUM+ZERO*TRACE(NHAF+I)
C   DECIMATE THE BAND LIMITED TRACE
    L=0
    DO 405 I=1,NPTS,INC
    L=L+1
405 TRACE(L)=WORK(I)
    NP=L
C   PUNCH THE BAND LIMITED, DECIMATED INPUT TRACE, FORMAT(5E15.7)
    CALL WRCP(IOOUT,TRACE,NP,1,1,50,IDATE,NAME,AUM)
    IPTS=NP+1
    LPTS=NP
    NPTS=NLEN
C   EXPAND TIME SCALE ON INPUT DATA AND OUTPUT DATA
    EXPAND=4.
```

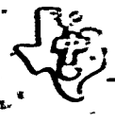
```
      KPTS=NPTS/EXPAND
      AELX=DELX*EXPAND
      AT=DT*INC
      WRITE(6,401) AT
  401 FORMAT(//,1X,'SAMPLE PERIOD OF BAND LIMITED AND DECIMATED DATA',
     1F10.2)
      DO 700 J=IPTS,NPTS
  700 TRACE(J)=AMAX
C     PLOT INPUT TRACE
      CALL LETTER(0.,0.,0.2,NAME,90.,60)
      YSIZE=2.
      IN=0
      IBOX=2
      XORIG=YSIZE+1.
      CALL ORIGIN(XORIG,0.)
      CALL LETTER(0.25,0.,0.15,'INPUT TRACE',90.,11)
      CALL PLOTA(TRACE,KPTS,AELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
      CALL WRCP(IDOUT,TRACE,NPTS,1,1,IDIS,IDATE,NAME,AUM)
      IDOUT=IDOUT+1
      TMAX=YMAX
      TMIN=YMIN
      DO 750 J=1,LPTS
      AJ=J
  750 WORK(J)=TRACE(J)*(ALPHA**AJ)
      DO 770 J=IPTS,NPTS
  770 WORK(J)=AMAX
C     STORE REAL INPUT INTO COMPLEX ARRAY X
      DO 8 J=1,NPTS
    8 X(J)=CMPLX(WORK(J),0.0)
C     TAKE FFT OF ARRAY X
      CALL NLOGN(X,NPTS,-1.0)
C     TAKE LOG OF X
      IEND=NPTS/2+1
      ISW=1
      REALP=REAL(X(1))
      IF(REALP.LT.0.) ISW=2
      GO TO (110,120),ISW
  120 DO 130 J=1,IEND
      REALP=-REAL(X(J))
      AI=-AIMAG(X(J))
  130 X(J)=CMPLX(REALP,AI)
      WRITE(6,140)
  140 FORMAT(1X,'CONSTANT PHASE COMPONENT REMOVED')
  110 CONTINUE
      DO 11 J=1,IEND
   11 X(J)=CLOG(X(J))
      REALP=REAL(X(IEND))
      X(IEND)=CMPLX(REALP,0.)
      RMAX=REAL(X(1))
      INDEXP=1.
      DO 12 J=1,IEND
      REALP=REAL(X(J))
      WORK(J)=REALP
      IF(REALP.LT.RMAX) GO TO 12
```

```
      RMAX=REALP
      INDEXP=J
   12 CONTINUE
      PLOT REAL PART OF LOG.
      YSIZE=7.
      IN=0
      IBOX=1
      XORIG=YSIZE+1.
      CALL ORIGIN (XORIG,0.)
      CALL LETTER(0.25,0.,0.15,'REAL PART OF LOG.',90.,17)
      CALL PLOTA(WORK,IEND,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
      PMAX=YMAX
      PMIN=YMIN
      WRITE(6,14) INDEXP,RMAX
   14 FORMAT(//,1X,'INDEX OF PEAK =',I10,10X,'MAX. VALUE, REAL PART OF L
     1OG. =',1PE15.7)
C     FIND INDEXES ABOUT THE PEAK FOR WHICH VALUES BETWEEN ARE ABOVE
C     THE CLIP LEVEL
      INDEXB=IEND
      INDEXA=1
      IF(DBLIM.LT.0.) GO TO 20
      CLIP=(1./(10**(DBLIM/20.)))
      CLIP=XLOG(CLIP)
      ACLIP=ABS(CLIP)
      CLIP=RMAX+CLIP
      IF(INDEXP.EQ.1) GO TO 17
      NTIMES =INDEXP-1
      DO 16 J=1,NTIMES
      K=INDEXP-J
      INDEXA=K
      IF((RMAX-WORK(K)).GE.ACLIP) GO TO 17
   16 CONTINUE
   17 CONTINUE
      IF(INDEXP.EQ.IEND) GO TO 20
      NTIMES=INDEXP+1
      DO 19 J=NTIMES,IEND
      IF((RMAX-WORK(J)).GE.ACLIP) GO TO 20
      INDEXB=J
   19 CONTINUE
   20 CONTINUE
      DO 22 J=1,IEND
   22 WORK(J)=AIMAG(X(J))
      GO TO (610,620),IAARON
  610 CONTINUE
      CALL AARON(X,WORK,COR,INDEXA,INDEXB,CLIP,NPTS,IEND,PI,TWOPI)
      GO TO 630
  620 CONTINUE
      CALL SCHAFR(X,WORK,COR,INDEXA,INDEXB,CLIP,NPTS,IEND,PI,TWOPI,TEMP)
  630 CONTINUE
      WRITE(6,21) INDEXA,INDEXB
   21 FORMAT(//,1X,'INDEXA =',I10,10X,'INDEXB =',I10)
C     ROTATE THE PHASE
      IF(INDEXB.EQ.IEND) INDEXB=INDEXB-1
      NPFIT=INDEXB-INDEXA+1
```

```
      DO 600 J=1,NPFIT
  600 TEMP(J)=J-1
      SLOPE=WORK(INDEXB)/TEMP(NPFIT)
      K=1
      DO 601 J=INDEXA,INDEXB
      SAVE(J)=SLOPE*TEMP(K)
      TEMP(J)=WORK(J)-SAVE(J)
  601 K=K+1
      DO 602 J=INDEXA,INDEXB
      REALP=REAL(X(J))
      COR(J)=COR(J)+SAVE(J)
  602 X(J)=CMPLX(REALP,TEMP(J))
      IP=0
      ICC=NPTS/2
      DO 51 J=2,ICC
      REALP=REAL(X(J))
      AI=-AIMAG(X(J))
      COR(NPTS-IP)=-COR(J)
      X(NPTS-IP)=CMPLX(REALP,AI)
   51 IP=IP+1
      IF(IAARON.EQ.2) GO TO 835
      PLOT REAL PART OF LOG. CLIPPED
      YSIZE=2.
      IN=1
      IBOX=1
      YMAX=PMAX
      YMIN=PMIN
      XORIG=YSIZE+1.
      CALL ORIGIN (XORIG,0.)
      CALL LETTER(0.25,0.,0.15,'REAL PART OF LOG. CLIPPED',90.,25)
      DO 54 J=1,NPTS
   54 WORK(J)=REAL(X(J))
      CALL PLOTA(WORK,NPTS,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
  835 CONTINUE
C     PLOT IMAG. PART OF LOG. CLIPPED AND CORRECTED
      YSIZE=2.
      IN=0
      IBOX=1
      XORIG=YSIZE+1.
      CALL ORIGIN(XORIG,0.)
      CALL LETTER(0.25,0.,0.15,'IMAG. PART OF LOG. CLIPPED AND CORRECTED
     1',90.,40)
      DO 55 J=1,NPTS
   55 WORK(J)=AIMAG(X(J))
      CALL PLOTA(WORK,NPTS,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
C     SAVE X ARRAY IN SAVE
      DO 200 J=1,NPTS
  200 SAVE(J)=X(J)
C     TAKE IFFT OF ARRAY X
      CALL NLOGN(X,NPTS,+1.0)
C     PLOT REAL PART OF IFFT (COMPLEX CEPSTRUM)
      YSIZE=2.
      IN=0
      IBOX=1
```

```
      XORIG=YSIZE+1.
      CALL ORIGIN(XORIG,0.)
      CALL LETTER(0.25,0.,0.15,'REAL PART OF IFFT (COMPLEX CEPSTRUM)',
     190.,36)
      PLOT CEPSTRUM FROM -W+1 TO +W
      JPTS=NRTS+1
      NHALF=NPTS/2
      DO 57 J=1,IEND
      WORK(J+NHALF-1)=REAL(X(J))
   57 STOR(J+NHALF-1)=WORK(J+NHALF-1)
      NHALF=NHALF-1
      DO 31 J=1,NHALF
      WORK(J)=REAL(X(IEND+J))
   31 STOR(J)=WORK(J)
      CALL PLOTA(WORK,NPTS,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
      CALL WRCP(IDOUT,WORK,NPTS,1,1,IDIS,IDATE,NAME,AUM)
      IDOUT=IDOUT+1
      DO 201 J=1,NPTS
      REALP=REAL(SAVE(J))
  201 X(J)=CMPLX(REALP,0.)
      SUMP=YMAX/10.
      SUMN=YMIN/10.
      DO 492 I=1,2
      DO 493 J=1,NPTS
      IF(STOR(J)) 494,493,495
  494 IF(STOR(J).LT.SUMN) STOR(J)=SUMN
      GO TO 493
  495 IF(STOR(J).GT.SUMP) STOR(J)=SUMP
  493 CONTINUE
      YSIZE=2.
      IN=0
      IBOX=1
      XORIG=YSIZE+1.
      CALL ORIGIN(XORIG,0.)
      CALL PLOTA(STOR,NPTS,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
      SUMP=SUMP/10.
      SUMN=SUMN/10.
  492 CONTINUE
      CALL NLOGN (X,NPTS,+1.0)
C     PLOT REAL PART OF IFFT OF REAL PART
      YSIZE=2.
      IN=0
      IBOX=1
      XORIG=YSIZE+1.
      CALL ORIGIN(XORIG,0.)
      CALL LETTER(0.25,0.,0.15,'R.P.IFFT.R.P.',90.,12)
      DO 203 J=1,NPTS
      TEMP(J)=REAL(X(J))
      STOR(J)=TEMP(J)
  203 WORK(J)=REAL(X(J))
      CALL PLOTA(WORK,NPTS,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
C     EXPANDED PLOT OF R.P.IFFT.R.P.
      SUMP=YMAX/10.
      SUMN=YMIN/10.
```

```
      DO 496 I=1,2
      DO 904 J=1,NPTS
      IF(STOR(J)) 903,904,905
  903 IF(STOR (J).LT.SUMN) STOR(J)=SUMN
      GO TO 904
  905 IF(STOR(J).GT.SUMP) STOR(J)=SUMP
  904 CONTINUE
      YSIZE=2.
      IN=0
      IBOX=1
      XORIG=YSIZE+1.
      CALL ORIGIN(XORIG,0.)
      CALL PLOTA(STOR,NPTS,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
      SUMP=SUMP/10.
      SUMN=SUMN/10.
  496 CONTINUE
      DO 205 J=1,NPTS
      AI=AIMAG(SAVE(J))
  205 X(J)=CMPLX(0.,AI)
      CALL NLOGN (X,NPTS,+1.0)
C     PLOT REAL PART OF IFFT OF IMAG PART
      YSIZE=2.
      IN=0
      IBOX=1
      XORIG=YSIZE+1.0
      CALL ORIGIN(XORIG,0.)
      CALL LETTER(0.25,0.,0.15,'R.P.IFFT.I.P.',90.,12)
      DO 207 J=1,NPTS
      WORK(J)=REAL(X(J))
  207 STOR(J)=WORK(J)
      CALL PLOTA(WORK,NPTS,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
C     EXPANDED PLOT OF R.P.IFFT.I.P.
      SUMP=YMAX/10.
      SUMN=YMIN/10.
      DO 497 I=1,2
      DO 914 J=1,NPTS
      IF(STOR(J)) 913,914,915
  913 IF(STOR(J).LT.SUMN) STOR(J)=SUMN
      GO TO 914
  915 IF(STOR(J).GT.SUMP) STOR(J)=SUMP
  914 CONTINUE
      YSIZE=2.
      IN=0
      IBOX=1
      XORIG=YSIZE+1.
      CALL ORIGIN(XORIG,0.)
      CALL PLOTA(STOR,NPTS,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
      SUMP=SUMP/10.
      SUMN=SUMN/10.
  497 CONTINUE
      CALL TTASK(T2)
      WRITE(6,8888) T2
 8888 FORMAT(//,1X,'TIME PART ONE ',F10.0,'  SECONDS')
      IF(IFILT.EQ.0) GO TO 999
```



```
      CALL STASK
      DO 68 J=1,NPTS
   68 SAVE(J)=CMPLX(TEMP(J),WORK(J))
C     READ FILTER DATA
C     FILTER THE CEPSTRUM
      DO 888 KFILT=1,IFILT
      DO 300 J=1,NPTS
      TEMP(J)=REAL(SAVE(J))
  300 WORK(J)=AIMAG(SAVE(J))
      READ(5,302) ITYPE,INITPT,IDX,JTAPER
  302 FORMAT(4I10)
      WRITE(6,320)
  320 FORMAT(//,1X,'ITYPE,INITPT,IDX,JTAPER')
      WRITE(6,321) ITYPE,INITPT,IDX,JTAPER
  321 FORMAT(1X,4I10)
      IFINPT=NPTS-INITPT+2
      GO TO (305,310,315),ITYPE
C     SHORT PASS SYSTEM
  305 DO 306 J=INITPT,IFINPT
      WORK(J)=0.
  306 TEMP(J)=0.
      IF(JTAPER.EQ.1) GO TO 316
      ITAPER=INITPT-4
      DEG=0.
      DO 665 J=ITAPER,INITPT
      ARG=DEG/57.2958
      ARG=(1.+COS(ARG))/2.
      WORK(J)=WORK(J)*ARG
      TEMP(J)=TEMP(J)*ARG
  665 DEG=DEG+45.
      ITAPER=IFINPT+4
      DEG=180.
      DO 667 J=IFINPT,ITAPER
      ARG=DEG/57.2958
      ARG=(1.+COS(ARG))/2.
      WORK(J)=WORK(J)*ARG
      TEMP(J)=TEMP(J)*ARG
  667 DEG=DEG+45.
      GO TO 316
C     LONG PASS SYSTEM
  310 DO 311 J=1,INITPT
      WORK(J)=0.
  311 TEMP(J)=0.
      DO 312 J=IFINPT,NPTS
      WORK(J)=0.
  312 TEMP(J)=0.
      IF(JTAPER.EQ.1) GO TO 316
      ITAPER=INITPT+4
      DEG=180.
      DO 666 J=1,INITPT,ITAPER
      ARG=DEG/57.2958
      ARG=(1.+COS(ARG))/2.
      WORK(J)=WORK(J)*ARG
      TEMP(J)=TEMP(J)*ARG
```

```
  666 DEG=DEG+45.
      ITAPER=IFINPT-4
      DEG=0.
      DO 668 J=ITAPER,IFINPT
      ARG=DEG/57.2958
      ARG=(1.+COS(ARG))/2.
      WORK(J)=WORK(J)*ARG
      TEMP(J)=TEMP(J)*ARG
  668 DEG=DEG+45.
      GO TO 316
C     COMB FILTER (INITPT=NUMBER OF NOTCHES)
  315 CONTINUE
      READ(5,318) (INOTCH(J),J=1,INITPT)
  318 FORMAT(8I10)
      DO 322 J=1,INITPT
      IDEX1=INOTCH(J)-IDX
      IDEX2=INOTCH(J)+IDX
      DO 323 K=IDEX1,IDEX2
      WORK(K)=0.
  323 TEMP(K)=0.
  322 CONTINUE
      IF(ITAPER.EQ.1) GO TO 316
  316 CONTINUE
      DO 317 J=1,NPTS
      REALP=TEMP(J)+WORK(J)
  317 X(J)=CMPLX(REALP,0.)
C     TAKE FFT OF ARRAY X
      CALL NLOGN(X,NPTS,-1.0)
C     REVERSE ROTATION
      DO 59 J=1,NPTS
      REALP=REAL(X(J))
      AI=AIMAG(X(J))+COR(J)
   59 X(J)=CMPLX(REALP,AI)
C     PLOT REAL PART OF SECOND FFT
      YSIZE=2.
      IN=1
      IBOX=1
      YMAX=PMAX
      YMIN=PMIN
      XORIG=YSIZE+1.
      CALL ORIGIN(XORIG,0.)
      CALL LETTER(0.25,0.,0.15,'REAL PART OF SECOND FFT',90.,23)
      DO 64 J=1,NPTS
   64 WORK(J)=REAL(X(J))
      CALL PLOTA(WORK,NPTS,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
C     TAKE EXPONENTIAL OF ARRAY X
      DO 60 J=1,NPTS
   60 X(J)=CEXP(X(J))
      IF(ITYPE.EQ.2) GO TO 150
      GO TO (150,160),ISW
  160 DO 170 J=1,NPTS
      REALP=-REAL(X(J))
      AI=-AIMAG(X(J))
  170 X(J)=CMPLX(REALP,AI)
```

```
  150 CONTINUE
C     TAKE IFFT OF ARRAY X
      CALL NLOGN(X,NPTS,+1.0)
C     PLOT REAL PART OF SECOND IFFT (OUTPUT )
      YSIZE=2.
      IN=1
      IBOX=2
      YMAX=TMAX
      YMIN=TMIN
      XORIG=YSIZE+1.
      CALL ORIGIN (XORIG,0.)
      GO TO (791,792,793),ITYPE
  791 CALL LETTER(0.25,0.,0.15,'REAL PART OF SECOND IFFT (SHORT PASS OUT
     1PUT)',90.,44)
      GO TO 794
  792 CALL LETTER(0.25,0.,0.15,'REAL PART OF SECOND IFFT (LONG PASS OUTP
     1UT)',90.,43)
      GO TO 794
  793 CALL LETTER(0.25,0.,0.15,'REAL PART OF SECOND IFFT (COMB OUTPUT)',
     190.,38)
  794 CONTINUE
      DO 62 J=1,NPTS
   62 WORK(J)=REAL(X(J))
      IF(ITYPE.EQ.2) GO TO 685
      DO 760 J=1,LPTS
      AJ=-J
  760 WORK(J)=WORK(J)*(ALPHA**AJ)
      GO TO 686
  685 IN=0
      K=1
      DO 687 J=1,LPTS
      IF(WORK(J).GT.0.5) GO TO 688
  687 K=K+1
      K=1
  688 CONTINUE
      WRITE(6,690) K
  690 FORMAT(//,1X,'INITIAL POINT FOR UNWEIGHTING LONG PASS OUTPUT =',
     1I10)
      L=LPTS+K-1
      M=0
      DO 689 J=K,L
      M=M+1
      AJ=-M
  689 WORK(J)=WORK(J)*(ALPHA**AJ)
  686 CONTINUE
      CALL PLOTA(WORK,KPTS,AELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
      CALL WRCP(IDOUT,WORK,NPTS,1,1,IDIS,IDATE,NAME,AUM)
      IDOUT=IDOUT+1
      IF(ITYPE.EQ.2) GO TO 888
C     PLOT TRACE (INPUT SIGNAL MINUS OUTPUT)
      YSIZE=2.
      IN=1
      IBOX=2
      YMAX=TMAX
```

```
      YMIN=TMIN
      XORIG=YSIZE+1.
      CALL ORIGIN (XORIG,0.)
      CALL LETTER(0.25,0.,0.15,'INPUT SIGNAL MINUS OUTPUT',90.,25)
      DO 81 J=1,NPTS
   81 TEMP(J)=TRACE(J)-WORK(J)
      CALL PLOTA(TEMP,KPTS,AELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,DT)
      CALL WRCP(IDOUT,TEMP,NPTS,1,1,IDIS,IDATE,NAME,AUM)
      IDOUT=IDOUT+1
  888 CONTINUE
      CALL TTASK(T2)
      WRITE(6,8889) IFILT,T2
 8889 FORMAT(//,1X,'TIME',I5,'  FILTERS  ',F10.0,'  SECONDS')
  999 CONTINUE
      CALL ORIGIN(4.,0.)
      CALL BLOKNO
 1000 CONTINUE
 1009 CONTINUE
      CALL ENDPLT
      CALL NN100
      STOP
      END
      SUBROUTINE PLOTA(DATA,NPTS,DELX,YSIZE,XLEN,YMAX,YMIN,IN,IBOX,INC,
     1DT)
      DIMENSION DATA(1)
      CALL BLOKNO
      XLEN=(NPTS-1)*DELX*INC*DT
      IF(IN.EQ.1) GO TO 10
      YMAX=DATA(1)
      YMIN=DATA(1)
      DO 1 J=2,NPTS
      IF(DATA(J).GT.YMAX) YMAX=DATA(J)
    1 IF(DATA(J).LT.YMIN) YMIN=DATA(J)
   10 SF=YSIZE/(YMAX-YMIN)
      IF(IBOX.EQ.1) CALL SQUARE(-YSIZE,0.,0.,XLEN)
      N=6
      DO 4 J=1,7
      IF(ABS(YMIN*10**N).LT.8400000.) GO TO 5
    4 N=N-1
      GO TO 6
    5 CALL RIGHTJ(0.,0.,0.1,YMIN,90.,N)
    6 IF(YMIN.GT.0..OR.YMAX.LT.0.) GO TO 2
      ZERO=YMIN*SF
      CALL RIGHTJ(ZERO,0.,0.1,0.0,90.,1)
      CALL PLOT(ZERO,0.,3)
      CALL DASH(ZERO,XLEN,0.1,0.1)
    2 N=6
      DO 7 J=1,7
      IF(ABS(YMAX*10**N).LT.8400000.) GO TO 8
    7 N=N-1
      GO TO 9
    8 CALL RIGHTJ(-YSIZE,0.,0.1,YMAX,90.,N)
    9 SFA=-SF
      AY=0.
```

```
      AX=(DATA(I)-YMIN)*SFA
      CALL PLOT(AX,AY,3)
      DO 3 J=2,NPTS
      AY=(J-1)*DELX*INC*DT
      AX=(DATA(J)-YMIN)*SFA
    3 CALL PLOT(AX,AY,2)
      RETURN
      END
      SUBROUTINE NLOGN (X,LX,SIGN)
C SUBROUTINE NLOGN COMPUTES THE DISCRETE FOURIER TRANSFORM BY THE
C N LOG N METHOD OR THE FAST FOURIER TRANSFORM METHOD
C
C
C THE ARGUMENTS ARE
C N    WHERE 2**N IS THE NUMBER OF TERMS IN THE X ARRAY
C X    THE ARRAY OF COMPLEX NUMBERS FOR BOTH INPUT AND OUTOUT
C SIGN EITHER -1.0 OR +1.0
C LX   COMPUTED AS 2**N
C NMAX=LARGEST VALUE OF N TO BE PROCESSED
C NONDUMMY DIMENSION M(NMAX)
C FOR EXAMPLE, IF NMAX=25 THEN
      DIMENSION M(25)
      DIMENSION X(1)
      COMPLEX X,WK,HOLD,Q
      FLX=LX
       ILK=1
       DO 3380 I=1,25
       IF (LX.LE.ILK) GO TO 3381
       N=I
 3380  ILK=ILK*2
 3381  CONTINUE
      DO 1 I=1,N
    1 M(I)=2**(N-I)
      DO 10 L=1,N
      NBLOCK=2**(L-1)
      LBLOCK=LX/NBLOCK
      LBHALF=LBLOCK/2
      K=0
      DO 11 IBLOCK=1,NBLOCK
      FK=K
      V=SIGN*6.2831853*FK/FLX
      WK=CMPLX(COS(V),SIN(V))
      ISTART=LBLOCK*(IBLOCK-1)
      DO 2 I=1,LBHALF
      J=ISTART+I
      JH=J+LBHALF
      Q=X(JH)*WK
      X(JH)=X(J)-Q
      X(J)=X(J)+Q
    2 CONTINUE
      DO 3 I=2,N
      II=I
      IF(K.LT.M(I)) GO TO 4
    3 K=K-M(I)
```

```
    4 K=K+M(II)
   11 CONTINUE
   10 CONTINUE
      K=0
      DO 7 J=1,LX
      IF(K.LT.J)GO TO 5
      HOLD=X(J)
      X(J)=X(K+1)
      X(K+1)=HOLD
    5 DO 6 I=1,N
      II=I
      IF(K.LT.M(I)) GO TO 7
    6 K=K-M(I)
    7 K=K+M(II)
      IF(SIGN.LT.0.0) RETURN
      DO 8 I=1,LX
    8 X(I)=X(I)/FLX
      RETURN
      END
      SUBROUTINE AARON(X,WORK,COR,INDEXA,INDEXB,CLIP,NPTS,IEND,PI,
     1TWOPI)
      DIMENSION X(1),WORK(1),COR(1)
      COMPLEX X
      ISUB=NPTS/2
      IF(INDEXA.EQ.1) GO TO 40
      WORK(1)=0.0
      X(1)=CMPLX(CLIP,WORK(1))
      WORK(2)=0.0
      X(2)=CMPLX(CLIP,WORK(2))
      GO TO 42
   40 CONTINUE
   42 CONTINUE
      COR(1)=0.
      COR(2)=0.
      DO 25 LL=3,ISUB
      IF(LL.LE.INDEXA) GO TO 30
      IF(LL.GT.INDEXB) GO TO 30
      IF(LL.LE.INDEXA+2) GO TO 410
      YHAT=2*WORK(LL-1)-WORK(LL-2)
      XMOD=AMOD(YHAT,TWOPI)
      IF(XMOD.GT.PI) XMOD=XMOD-TWOPI
      IF(XMOD.LE.-PI) XMOD=XMOD+TWOPI
      PHA=WORK(LL)
      DIFF=ABS(XMOD-PHA  )
      SIGN=+1.
      IF(PHA  -XMOD.LT.0.) SIGN=-1.
      IF(DIFF.LT.PI) WORK(LL)=YHAT+PHA  -XMOD
      IF(DIFF.GT.PI) WORK(LL)=YHAT+PHA  -XMOD-(TWOPI*SIGN)
      COR(LL)=PHA  -WORK(LL)
      RX=REAL(X(LL))
      X(LL)=CMPLX(RX,WORK(LL))
      GO TO 25
  410 COR(LL)=0.
      RX=REAL(X(LL))
```

```
      X(LL)=CMPLX(RX,WORK(LL))
      GO TO 25
   30 WORK(LL)=0.0
      COR(LL)=0.
      X(LL)=CMPLX(CLIP,WORK(LL))
   25 CONTINUE
      IF(INDEXB.LE.ISUB) X(IEND)=CMPLX(CLIP,WORK(IEND))
      COR(IEND)=0.
      RETURN
      END
      SUBROUTINE SCHAFR(X,WORK,COR,INDEXA,INDEXB,CLIP,NPTS,IEND,PI,
     1TWOPI,TEMP)
      DIMENSION X(1),WORK(1),COR(1),TEMP(1)
      COMPLEX X
      ISUB=NPTS/2
      EP=4.8
      EP=1.
      EP=TWOPI-EP
      INDEXA=1
      INDEXB=IEND
C     THIS ROUTINE ALLOWS NO CLIPPING
      COR(1)=0.
      TEMP(1)=0.
      DO 15 J=2,ISUB
      COR(J)=COR(J-1)
      DIFF=WORK(J)-WORK(J-1)
      IF(DIFF.GT.EP ) COR(J)=COR(J-1)-TWOPI
      IF(DIFF.LT.-EP) COR(J)=COR(J-1)+TWOPI
   15 TEMP(J)=WORK(J)+COR(J)
      DO 20 J=1,ISUB
      COR(J)=-COR(J)
      REALP=REAL(X(J))
      WORK(J)=TEMP(J)
   20 X(J)=CMPLX(REALP,TEMP(J))
      COR(IEND)=0.
      RETURN
      END
```